"It means we are weakening the dollar. We are trying to liquidate our debt through inflation. The consequence of what the Fed is doing is a lot more than just CPI. It has to do with malinvestment and people doing the wrong things at the wrong time. Believe me, there is plenty of that. The one thing that [Ben] Bernanke has not achieved and it frustrates him, I can tell – is he gets no economic growth. He doesn't do anything with the unemployment numbers. I think the country should have panicked over what the Fed is saying that we have lost control and the only thing we have left is massively creating new money out of thin air, which has not worked before, and is not going to work this time."

––– September 13, 2012 quote from Congressman Ron Paul (R–TX) on Boomberg Television's "Street Smart" discussing the Federal Reserve causing inflation by creating even more money out of thin air ($40 billion a month now) via "quantitative easing."

(realclearpolitics.com/video/2012/09/13/ron_paul_fed_creating_new_money_out_of_thin_air.html)

Table of Contents

235-600-700   **December 2003**

1. INTRODUCTION

1.1 PURPOSE

The *5ESS*® Switch Input Messages document describes the input messages (IMs) available for use on the *5ESS*® switch input channels. This document is a reference guide for *5ESS*® switch support personnel. This document is available in electronic media only.

The messages in this document and its updates represent complete documentation for the 5E14 and later software releases.

The *5ESS*® switch IMs are used to control, maintain, and monitor the switching system, including the processors, peripherals, and other software. For example, the messages can direct the system to:
- Execute diagnostic and exercise programs and report the results.
- Perform tests and report the results.
- Report the status of various subsystems.
- Report traffic information.
- Enter translation information into memory.
- Restore units to service or remove units from service.

Because of the many available options, some IMs are complex. Before using these particular messages, become familiar with conventions used in this document to describe them (refer to the User Guidelines section).

Some messages may adversely affect service. These messages show a **WARNING** appearing opposite the key block. Use these messages only during periods of light traffic. Before using one of these messages, read thoroughly the manual page containing the message description.

1.1.1 UPDATE INFORMATION

This document is being updated to include documentation for the 5E17(1) software release of the *5ESS*® switch. It also includes numerous enhancements AND responses to customer requests for more information.

The following messages/appendixes have been updated:

```
ALW:FAC-C
CFR:SPRMEM-E
EXC:PM-D
INH:FAC-C
INIT:FAC-C
OP:FAC-C
OP:ST-GQPHLNK
OP:ST-GQPHPIPE
OP:ST-SERV
RMV:GQPHPIPE
RST:GQPHPIPE
ST:NIPMP-E
SW:SERV
TRC:UTIL-C
TRC:UTIL-D
TRC:UTIL-E
TST:PATH-C
```

1.1.1.1 SUPPORTED SOFTWARE RELEASES

2

235-600-700   December 2003

In accordance with the *5ESS*® Switch Software Support Plan, the 5E13 software release is rated Discontinued Availability (DA) as of August 30, 2002. The information supporting 5E13 and earlier is being removed over time, instead of concurrently, from all documentation.

If you are supporting offices that use a software release prior to 5E14 and you have a need for the information that is being removed, retain the associated pages as they are removed from the paper documents, or retain the earlier copy of the CD-ROM.

The purpose of this documentation is to facilitate early dissemination of information. Its contents are subject to change pursuant to the general non-disclosure agreements between Lucent Technologies and *5ESS*® switch owners for the purpose of planning. To the best of Lucent Technologies's knowledge, the information contained in this document is accurate and complete as of the date of publication. HOWEVER, LUCENT TECHNOLOGIES EXPRESSLY DISCLAIMS ANY WARRANTY AS TO ACCURACY OR COMPLETENESS NOR DOES LUCENT TECHNOLOGIES ASSUME ANY RESPONSIBILITY FOR THE USE OF THE INFORMATION BY OTHERS. Lucent Technologies reserves the right to change or delete any portions of the document or to add information in the future.

1.1.1.2 TERMINOLOGY

1.1.1.2.1 Communication Module Name Change

Global Messaging Server (GMS) is the official name of the communication module, model 3 (CM3) hardware. Where the term GMS may be expected in software-influenced items such as input and output messages, master control center screens, and recent change/verify screens, the term CM3 may be found until such time as the term is changed in the software code.

1.1.1.2.2 Bellcore/Telcordia Name Change

As of March 18, 1999, Bellcore officially changed its name to Telcordia Technologies. Not all pages of this document are being reissued to reflect this change; instead, the pages will be reissued over time, as technical and other changes are required. Customers on standing order for this document may see that, on previous-issue pages, the Bellcore name is still exclusively used.

Customers receiving new orders for this document will see the Telcordia Technologies name used as appropriate throughout the document, and the Bellcore name used only to identify items that were produced under the Bellcore name. Exceptions may exist in software-influenced elements such as input/output messages, master control center screens, and recent change/verify screens. These elements will not be changed in this document until such time as they are changed in the software code. Document updates will not be made specifically to remove historical references to Bellcore.

1.1.1.2.3 *5ESS*®-2000 Switch Name Change

This *5ESS*® switch document may contain references to the *5ESS*® switch, the 5ESS-2000 switch, and the *5ESS*® *AnyMedia*® Switch. The official name of the product has been changed back to the *5ESS*® switch. The documentation will not be totally reissued to change these references. Instead, the changes will be made over time, as technical changes to the document are required. In the interim, assume that any reference to the 5ESS-2000 switch or the *5ESS*® *AnyMedia*® Switch is also applicable to the *5ESS*® switch. It should be noted that this name change may not have been carried forward into software-influenced items such as input and output messages, master control center screens, and recent change/verify screens.

1.1.1.2.4 Document Specific Terminology

National ISDN is an evolving platform in which new features will continue to be introduced for new revenue opportunities, improved operational efficiencies, and for support of specific applications. NI 1, NI 2, and NI 3

235-600-700   December 2003

represent specific features as documented in Bellcore SRs 1937, 2120, and 2457. The industry is migrating to an additional terminology to more specifically denote the availability of National ISDN features: NI 95, NI 96, etc. A feature is included in a specific version (such as, NI 96) if it is available by the switch vendors by the first quarter of the year.

As a result of the World Telecommunications Standardization Conference held March 1-12, 1993, the International Telegraph and Telephone Consultative Committee (CCITT), no longer exists as an organization under the International Telecommunication Union (ITU). According to the ITU, the CCITT is now referred to as the International Telecommunication Union - Telecommunication Standardization Sector (ITU-TS).

For new and revised Recommendations issued by the ITU-TS, the term "CCITT Recommendation X.xxx" will be replaced by the "ITU-T Recommendation X.xxx" designation. For a transition period from 1993 to 1997, if the Recommendation had a previous CCITT designation, the new name will include "(formerly CCITT Recommendation X.xxx)". Names of existing CCITT Recommendations will not change unless revised.

1.1.1.3 HANDLE DATE AND TIME TRANSITION TO YEAR 2000

For any time-stamp data that occurs in *5ESS*® switch operations after the time is transitioned to the year 2000 and beyond, the year 00 shall be interpreted as 2000 after 12/31/1999. The data (billing, log files, and so forth) shall continue to work even right after transition to the year 2000 from midnight 12/31 1999.

1.1.2 ORGANIZATION

The IM manual includes the:

Introduction   - explains the purpose of the document and the organization of the material presented.

User Guidelines - contains information about the conventions used in documenting IMs. The structure and layout of IMs, as well as their naming and usage conventions, are discussed. This section also explains the on-line help feature, the input message edit and history function, the format notation used in this document, and the organization and content of the message descriptions contained in this document.

Acronym List   - defines acronyms and abbreviations used in the IM manual.

Indexes   - contains helpful indexes. These indexes may be used as guides for locating and selecting specific message descriptions in the IM manual.

- The Functional Index summarizes the purpose of each IM, categorizing the IM according to unit or function.

- The Topical Index shows the relation of IMs divided into topics (categories).

Appendixes   - contains information that is referred to by two or more IM descriptions, such as listings of audits and traffic (TRFC) sections.

The rest of the IM manual contains message descriptions presented in alphabetical order by message ID.

1.2 USER COMMENTS

We are constantly striving to improve the quality and usability of this information product. Please use one of the following options to provide us with your comments:

- You may use the on-line comment form at **http://www.lucent-info.com/comments**

- You may email your comments to **comments@lucent.com**

Please include with your comments the title, ordering number, issue number, and issue date of the information product, your complete mailing address, and your telephone number.

If you have questions or comments about the distribution of our information products, see Section 1.3 , Distribution.

1.3 DISTRIBUTION

For distribution comments or questions, contact your local Lucent Technologies Account Representative.

A documentation coordinator has authorization from Lucent Technologies to purchase our information products at discounted prices. To find out whether your company has this authorization through a documentation coordinator, call **1-888-LUCENT8 (1-888-582-3688)**.

Customers who are not represented by a documentation coordinator and employees of Lucent Technologies should order *5ESS*® switch information products directly from Lucent Technologies.

To order, call the following telephone number:

- **1-888-LUCENT8 (1-888-582-3688)** or fax to **1-800-566-9568**; from inside the continental United States

- **1-317-322-6416** or fax to **1-317-322-6699**; from outside the continental United States.

1.4 TECHNICAL ASSISTANCE

For technical assistance, call Technical Support Services (TSS) at:

- **1-866-LUCENT8 (1-866-582-3688)**; from inside the continental United States

- **1-630-224-4672**; from outside the continental United States.

Technical Support Services is staffed 24 hours a day, 7 days a week.

1.5 SECURED/PROPRIETARY FEATURES

For all secured features, a right to use (RTU) fee must be paid before enabling information is provided. There may be additional proprietary documentation needed to interpret information regarding these features. Contact your Account Team Representative for additional information.

For a complete list of secured feature identifiers (SFID), refer to the *Translation Guide (TG-5)* manual, 235-080-100.

1.6 REFERENCES

This is a standard *5ESS*® switch document that is also applicable to the Compact Digital Exchange (CDX) and Very Compact Digital Exchange (VCDX) switching systems. Information applicable to only CDX or VCDX may be found in the following documents:

235-120-010   Compact Digital Exchange (CDX) Reference Guide

235-120-020   Compact Digital Exchange (CDX) User's Guide

235-120-120   Very Compact Digital Exchange (VCDX) User's Guide

Additional references are given as needed, within individual message descriptions. An "x" or "x"s in the last three positions of a release specific document number indicate the digits that change from release to release. Refer to 235-001-001, Documentation Description and Ordering Guide for the document number associated with each software release.

5ESS Input Messages – Part 1

235-600-700   December 2003

2. USER GUIDELINES

2.1 INTRODUCTION

This section explains the rules to which 5ESS® switch input messages (IMs) must conform, and the conventions that are used in this manual to show how messages are constructed.

The term messages refers to the commands that are used to control and monitor the 5ESS® switch system (IMs).

Each 5ESS® switch system is set to accept only one message language, human-machine language (MML). MML is the human interface language developed by the International Telecommunication Union - Telecommunication Standardization Sector (ITU-TS) (formerly CCITT).

2.2 MESSAGE DESCRIPTIONS

The message descriptions form the bulk of the IM manual. Message descriptions are documented on manual pages.

The term "manual page" refers to the description of an IM without regard to how many physical sheets of paper or screens are used to complete the description.

Each IM manual page consists of several elements that form the message description.

2.3 MANUAL PAGE - GENERAL PARTS

This section gives a general description of each part of a manual page.

2.3.1 Page Headers (Paper Only)

The header consists of two lines across the top of each manual page. The header on the inside of the manual page identifies the document number and the issue date. The header on the outside of the manual page shows the name of the document (for example, 5ESS® Switch Input Messages), and the message name.

2.3.2 Message Name

The message name is made up of the command code and parameter names from the message format omitting any variables and optional values and has a maximum length of 20 characters. This is used like a dictionary heading to give a general idea of placement in the manual.

2.3.3 Page Footers (Paper Only)

The footer on the inside of the page contains the issue number. The outside footer shows the message ID and page number. The message ID is explained with the key block items.

2.3.4 The Key Block

The key block is the block of text in the upper right corner of the manual page. The key block includes:
* ID
* RELEASE
* COMMAND GROUP
* APPLICATION

2.3.4.1 ID

Page 1

The ID is the unique identifier of the IM. Every message ID is composed of two parts separated by a colon. The part to the left of the colon is the command code of the message. One or more parameter names from the format usually make up the remaining portion of the ID. If there is more than one parameter name to the right of the colon, these parameter names are separated by a hyphen. For example: EXC:SODD-RED-OP.

2.3.4.2 RELEASE

This was formerly labeled GENERIC. The *5ESS*® switch software release or range of releases to which the message description applies. Here are some examples of possible software release entries:

5E14 only	The message description applies only to release 14.
5E14 and later	The message description applies to release 14 and to all later releases.
5E14 - 5E15	The message description applies to release 14 through release 15.

If changes from one release to another have affected a format, two or more separate manual pages are provided. Such a split is indicated by the last character of the message ID. If separate manual pages exist for the same message, the ID for the first manual page is followed by "-A", the second by "-B", and so forth.

Please note that this "dash-letter" suffix is dynamic depending on thes number of software releases being supported and the number of versions of a message.

2.3.4.3 COMMAND GROUP

The command group identifies the message as a member of a group of input messages having a certain authority/priority level on the switch. This code can sometimes be changed by the service provider. What is documented on the manual page is the command group originally assigned by AT&T in the initial software release.

2.3.4.4 APPLICATION

This specifies the application to which the message pertains.

Application Code	Explanation
5	USA *5ESS*® switch. May also apply to CDX, VCDX, or WNC. Refer to the help facility to verify applicability for these systems.
3B	administrative module (AM) software.
CNI	Common Network Interface.
CDX	Compact Digital Exchange only.
VCDX	Very Compact Digital Exchange. May also apply to the AEWNC. Refer to the help facility to verify applicability for this system.
AEWNC	*Air Extension*^SM Wireless Network Controller only.

This implies that a listing such as "5,3B" means that this message applies to both USA *5ESS*® switch and the administrative module software.

2.3.5 WARNING

Input messages that may adversely affect service are flagged with a warning message at the left side of the key block. An explanation of the warning is given in the PURPOSE section.

2.3.6 PURPOSE

This section of the manual page contains a brief explanation of the purpose of the message and includes the explanation of any associated warnings.

2.3.7 FORMAT

Message formats are a kind of notation that is used in this document to show the possible ways in which IMs can be constructed. Formats are described in greater detail in the **MANUAL PAGE-SECTION DETAILS** section of the User Guidelines.

2.3.8 EXPLANATION OF MESSAGE

This section of the manual page explains the meaning of the various parameter names, parameter values, and variables in the format.

2.3.9 SYSTEM RESPONSE

The system responds to IMs with an acknowledgment that appears one space after the terminating character of the IM on the TTY device. This will normally happen about five seconds after the message is entered.

These acknowledgments give information about the status of the message (for example, accepted, rejected, in progress, and so forth). Standard system responses are listed in the APP:RESPONSES appendix in the Appendixes section of the Input Messages manual. Standard craftshell responses are listed in Table 2-1 at the end of the User Guidelines. If a message has a response that is not standard, the response is explained in this section of the manual page.

2.3.10 REFERENCES

This section lists all related input and output messages, documentation references, Master Control Center (MCC) display pages and recent change views.

2.4 MANUAL PAGE - SECTION DETAILS

2.4.1 FORMAT Section

The format is the most important part of every manual page. The FORMAT shows the syntax of the message (that is, the way the message must be constructed so that the system will understand it correctly).

2.4.1.1 Example

Here is an example of how the FORMAT section of a typical manual page might look:

```
[1]      OP:PMCR,TYPE=STAT[,HOURLY|,DAILY=a|,NOHOURLY|,NODAILY];

[2]      OP:PMCR,TYPE=RET,INTVL=b,FORM=c;

[3]      OP:PMCR,TYPE=DEM,FORM=c;
```

Many IMs can be input in more than one format. When appropriate, distinct formats are shown separately and given a number (in square brackets). In the example, there are three formats. (The OP:PMCR message requests various plant measurements detailing system performance statistics.)

Format 1 is a status report request. It reports hourly and/or daily data.

Format 2 is a retained report request. It reports hourly and/or daily data that was retained from a previous status report.

Format 3 is a demand report request. It reports up-to-the-hour data collected for the next daily report.

2.4.1.2 Format Notation

The message formats shown in this document may include one or more types of format notation that are not part of the message itself. The purpose of this notation is to show the choices the user has in entering the message.

[ ]	Brackets enclose optional entries. Optional entries may be included in the message or may be omitted.
{ }	Braces enclose two or more entries where one entry, but only one, must be included in the message.
|	OR bars separate a selection of entries enclosed by braces or brackets. One of the entries separated by OR bars, but only one, may be selected.
. . .	Vertical ellipsis marks (three vertical dots) following a format means that there may be more than one line when an OM is printed. These additional lines will conform to the same format, but may contain different data.
...	Horizontal ellipsis marks used within a format depict variable repetition.
. . .	Three ellipsis marks with embedded spaces, either at the beginning of a line or at the end of a line, indicate that a format string is divided into multiple lines in the manual page. This is for documentation purposes only.

Remember that brackets, braces, OR bars, and ellipsis marks are never used when you are actually entering a message. They are only used in message formats to show you how a message must be constructed.

2.4.1.3 Notation Examples

The following examples illustrate the use of brackets, braces, OR bars, and ellipsis.

`ENTRY1 ENTRY2`	Both ENTRY1 and ENTRY2 must be used.
`[ENTRY1][ENTRY2]`	Both entries are optional. Both, either, or neither may be used.
`{ENTRY1|ENTRY2}`	Either ENTRY1 or ENTRY2 must be used. Using both is not permitted.
`[ENTRY1|ENTRY2]`	Either ENTRY1 or ENTRY2, but not both, may be used. Both entries may be omitted.
`[ENTRY1{ENTRY2|ENTRY3}]`	All three entries may be omitted; however, if any of these entries is used, your options are: ENTRY1 and ENTRY2 or ENTRY1 and ENTRY3. It is not permissible to use both ENTRY2 and ENTRY3.
`{ENTRY1|ENTRY2|ENTRY3}`	One of these three entries must be used, but it is not permissible to use more than one.
`[ENTRY1]...[ENTRY1]`	Signifies repetition of ENTRY1.

`a  b  c`
`.`
`.`
`.`
 	
`.`
`.`
`.`
 	
`.`
`.`
`.`
	Signifies that there may be more than one line when an OM is printed.

2.4.1.4 Special Conventions

A special format convention governs cases where a colon is followed by a series of consecutive optional entries separated by commas. In these cases, the colon replaces the comma for the first optional entry that is actually used. For example, DGN:MTC is documented with the following format:

```
DGN:MTC=a[:[RPT=b][,RAW][,UCL][,REX|,DEX]][,[PH=c[&&d]][,TLP]
        [,CONT][,MT=e]];
```

The following examples show the punctuation of this message when some (but not all) of the optional entries are chosen.

```
DGN:MTC=a:RPT=b,RAW,UCL,REX,PH=c,TLP;
DGN:MTC=a:RAW,UCL,REX,PH=c,TLP;
```

```
DGN:MTC=a:UCL,REX,TLP;
DGN:MTC=a;
```

2.4.2 Message Syntax

The structure of messages consists of a series of fields that must be entered in a prescribed order. Fields and the entries that can be made in them are summarized in Table 2-2 of the User Guidelines.

This is the general syntax of a message:
```
command code:parameter block(s):...:[data parameters block(s)];
```

IMs consist of two or more fields. The first field of an IM always contains a command code that identifies the action that the message is being used to accomplish. This command code is always followed by a colon.

The colon is followed by one or more parameter blocks. A parameter block consists of parameter names with associated parameter values. Parameter blocks are separated by colons. Parameter blocks identify units, functions, or data that the command code is acting upon.

A parameter name may appear by itself, or it may be followed by a single parameter value, a list of values, or a range of values. Parameter names are alphanumeric and must begin with a letter.

A parameter value is separated from a parameter name by an equal sign (=). Parameter values may be numeric or alphanumeric. When parameter values are shown in the FORMAT section of the manual page as lowercase letters, they represent variables that are replaced by specific data when the message is entered. If the value of the variable contains lowercase letters, the value must be enclosed in double quotes to retain the integrity of the lowercase letters.

Values in a list are separated by hyphens. Any two values that indicate the limits of a range are separated by two ampersands (&&), instead of a hyphen.

2.4.2.1 Syntax Examples

The following examples illustrate command codes and parameter blocks in various IMs.

`CLR:UMEM;`	Two fields: the command code (CLR), and one parameter block (UMEM).
`ABT:TASK:TLWS=a;`	Three fields: the command code (ABT), and two parameter blocks (TASK and TLWS=a)
`UPD:GEN:BACKOUT:UCL;`	Four fields: the command code (UPD), and three parameter blocks (GEN, BACKOUT, and UCL).
`OP:STATUS,PROCESS,ALL;`	Three fields: the command code (OP) and two parameter blocks (STATUS and PROCESS,ALL). NOTE: A comma is used here instead of the colon separating STATUS and PROCESS,ALL.

2.4.2.2 Parameter Name and Parameter Value Examples

The following examples show instances of parameter names and parameter values. `OP:STATUS,DN=a;`

The purpose of this message is to output the current status history for a specified line. The IM has the command code of OP (for "output"), and the parameter names STATUS and DN in the next field. The parameter name DN has the parameter value 'a'. This value stands for the directory number of the line for which the status history is to be printed. If you were to type in this message, you would substitute an actual directory number for the value 'a'. `SET:CLK,DATE=a-b-c,TIME=d-e-f;`

This message sets the system clock to the date and time that you specify. There are three names in the second field

(CLK, DATE, and TIME). If you were to type in the command, you would enter the month for 'a', the day of the month for 'b', and the year for 'c'. Similarly, you would enter the hour, minute, and second for values 'd', 'e', and 'f'.
`CLR:ISOL,SM=a[&&b];`

This message takes one or more switching modules (SMs) out of isolation. The parameter name SM has one or two values ('a' and 'b'). You must enter at least one. If you choose to enter only 'a', then you will enter only one SM number. If you choose the option, then you will enter a range of SM numbers. In this case, 'a' will stand for the lower limit of the range, and 'b' stands for the upper limit of the range. (Note the two ampersands that precede the 'b'. These ampersands indicate that a range is being specified.)

2.4.2.3 Comparison of Format Syntax and Actual Input Messages

The following examples show message format syntax as they might appear in manual pages and some examples of actual IMs that correspond to these formats. The FORMAT section of a manual page uses uppercase to indicate command codes and parameter names. Lowercase letters are used to indicate variables. In actual use at the user input terminal, all command codes and parameter names may be typed in lowercase.

Syntax:
Example:
or
	`OP:STATUS,DN=a;`
	`OP:STATUS,DN=19;`
	`op:status,dn=19`

Syntax:
Example:
or
	`SET:CLK,DATE=a-b-c,TIME=d-e-f;`
	`SET:CLK,DATE=12-22-94,TIME=2-30-15;`
	`set:clk,date=12-22-94,time=2-30-15;`

Syntax:
Examples:
or
 	
or
	`CLR:ISOL,SM=a[&&b];`
	`CLR:ISOL,SM=5&&12;`
	`clr:isol,sm=5&&12;`
	`CLR:ISOL,SM=9;`
	`clr:isol,SM=9;`

Syntax:
Examples:
or
 	
or
 	
or
	`OP:STATUS,PROCESS,{ALLTERMS|ALLKERNS|ALL}[,OPL=a];`
	`OP:STATUS,PROCESS,ALLTERMS,OPL=10;`
	`op:status,process,allterms,opl=10;`
	`OP:STATUS,PROCESS,ALL,OPL=10;`
	`op:status,process,all,opl=10;`
	`OP:STATUS,PROCESS,ALL;`
	`op:status,process,all;`

Syntax:
Examples:
or
 	
or
 	
or
 	
or
	`IN:REMOTE,START[,BSDIR="a"][,BLOCKS=b];`
	`IN:REMOTE,START,BSDIR="/etc/bwm",BLOCKS=100;`
	`in:remote,start,bsdir="/etc/bwm",blocks=100;`
	`IN:REMOTE,START,BSDIR="/etc/bwm";`
	`in:remote,start,bsdir="/etc/bwm";`
	`IN:REMOTE,START,BLOCKS=100;`
	`in:remote,start,blocks=100;`
	`IN:REMOTE,START;`
	`in:remote,start;`

2.4.2.4 Entering an IM

The craftshell accepts simplified IM syntax. Commas rather than colons may be used to separate the parameter blocks after the command code and first colon. All options may be specified as a single, comma-separated parameter block.

Parameter names may be typed in any order within a parameter block, but the parameter blocks must appear in the order specified in the manual page.

2.4.2.5 Syntax Combination Example

This example may be entered in a variety of format syntax combinations:
`DGN:IOP=1:RPT=1,RAW,UCL:DATA,CONT,TLP;`

Valid examples:

`DGN:IOP=1,RPT=1,RAW,UCL:DATA,CONT,TLP;`	(Colon between IOP and RPT has been changed to a comma.)
`DGN:IOP=1,RAW,UCL,RPT=1:DATA,CONT,TLP;`	(Order of keywords in the third block is rearranged.)
`DGN:IOP=1,RAW,UCL,RPT=1:DATA,TLP,CONT;`	(Order of keywords in the fourth block is rearranged.)
`DGN:IOP=1,RAW,UCL,RPT=1,TLP,CONT;`	(The ":" has been changed to a comma and the "`DATA,`" has been deleted.)

Invalid examples:

`DGN,IOP=1,RAW,UCL,RPT=1,CONT,TLP;`	(First colon has been changed to a comma.)
`DGN:IOP=1,CONT,TLP,RAW,UCL,RPT=1;`	(Order of blocks 3 and 4 reversed.)
`DGN:IOP=1,RAW,UCL,CONT,TLP,RPT=1;`	(Parameters of blocks 3 and 4 are intermixed.)

Because of the way that IMs are translated internally by the system, the message that is echoed to an output device may look different from the message originally entered at the console.

For example, the actual message: `DGN:IOP=1:RPT=1,RAW,UCL:DATA,CONT,TLP;` may be echoed to an output device from any of the following IM formats:

`DGN:IOP=1,RPT=1,RAW,UCL,CONT,TLP;`

`DGN:IOP=1,RPT=1,UCL,RAW,CONT,TLP;`

`DGN:IOP=1,UCL,RAW,RPT=1,CONT,TLP;`

`DGN:IOP=1,UCL,RAW,RPT=1,TLP,CONT;`

2.4.2.6 A Summary of Syntax Rules

A summary of rules for constructing IMs follows:

(1)   Block separators (:) may be typed as commas (,) except for the colon immediately following the command code.

(2)   Parameter names may be unordered within a parameter block, but the blocks themselves must remain ordered.

(3)   All numbers in IMs are assumed to be decimal unless you indicate otherwise by entering the appropriate code before the number. Valid codes are:

- B' (binary)

- O' (octal)

- H' or X' (hexadecimal)

- D' (decimal)

(4)   An entered message may consist of up to 255 spaces and characters that may extend over several lines of input.

(5)   The last character in a format line must be either an exclamation point, a semicolon, or a question mark.

- The semicolon (provided automatically by the return key) causes the message to be executed immediately as entered.

- The exclamation point is a continuation character.

- The question mark requests craftshell help.

(6) Within a message, the equal sign (=) is used as a separator between the parameter name and the parameter value. Spaces are not allowed between parameters.

(7) Double quotation marks ('' '') around a parameter value are required when specifying a pathname to a file or file name or to maintain the integrity of lowercase letters.

(8) Typing a non-printable character in a string results in ?(BEL) (a question mark and a bell sound). The non-printable character remains in the string.

2.5 ROUTING INPUT MESSAGE RESPONSES TO THE RECEIVE-ONLY PRINTER (ROP)

An output message response to an IM request is always printed on the originating TTY. The output message may also print on the receive-only printer (ROP) and/or other I/O devices, depending on the message class associated with the output message. There may be message classes that have an ''RSP'' or ''_RSP'' suffix (shown in the key block section of an output message description). These types of messages print at the originating TTY terminal only, in ''response'' to the input request.

Output messages with an ''RSP'' or ''_RSP'' message class can be forced to print at the ROP by appending '',PRTROP'' to the end of the IM request. The '',PRTROP'' option is valid on any IM and does not adversely affect the routing of the output message, except to route it to the ROP.

2.6 HELP FEATURE

The *5ESS*® switch has an IM HELP feature. This feature:

- Improves the understanding of error messages in cases where syntax or semantic errors are found in entered messages.

- Helps with IM syntax, including parameter value type and range.

- Prompts for entering IMs.

NOTE: Do not use HELP while the emergency action interface (EAI) page is displayed. Instead, press the NORM/DISP key to display a non-EAI page, then use HELP. If HELP is used while the EAI page is displayed, only the bottom few lines of the HELP message are displayed. The EAI page may also be left partially blank, but can be restored manually by pressing the EA/DISP key.

2.6.1 The Different Kinds of Help

There are two different ways to get help. One way is to type a portion of an IM, followed by a question mark. Another way is to type a question mark after an error message. Either of these will get you help.

There are two different types of help. The first type of help provides information about the syntax of IMs. The second type of help assists you in composing IMs by prompting choices between the possible entries.

2.6.2 Getting Help If You Know One or More Parameter Name

If you know one or more of the parameter names in a message, you can find out how many messages share that parameter, and what the syntax of these messages is.

To get information about the parameter names in an IM, enter part of the message (one or more parameter names). Type a question mark after the last parameter name.

```
EXAMPLE 1
If you enter:            op:pmcr?
The system responds:
                        The input matches 3 messages.
                        Further errors may be revealed after a single
                        message has been selected.

                        1) OP:PMCR:TYPE=a,FORM=b;

                        2) OP:PMCR:TYPE=a,INTVL=b,FORM=c;

                        3) OP:PMCR:TYPE=a[,b][,FORM=c];
                             b is HOURLY, DAILY[=d], NOHOURLY or NODAILY;
                             choose one or omit

                        append text, hit ? for more info.,
                        hit <CR> to execute, or hit <DEL> to reject

                        op:pmcr
```

These are the only messages that begin with "OP:PMCR". Up to 128 messages can be displayed. The underline character in the last line of the example represents the position of your cursor after the system has displayed its response on your terminal. The system will wait about forty seconds for your response. If you do not respond within that time, then you are exited from the help facility, and you get the system prompt.

You may do one of the following:

- Enter a carriage return to execute, "OP:PMCR" which results in a missing parameter name errors message.

- Type the rest of a message [for example, ":type=dem,form=1" (for format 1)] and a carriage return to execute the message.

- Type a question mark to go to the second help level. You are then taken through the message a step at a time, and you are prompted for each entry.

- Press the "Delete" key. You leave the help facility and the message is not sent.

NOTE: If you type a question mark immediately after a parameter name (as in the example), you get information only about the parameter names in the message. You do not get information about parameter values.

You may also get help for parameter values in a message. To get help for parameter values in a message, type an equal sign (=) followed by a question mark after the parameter name.

```
EXAMPLE 2
If you enter:            op:conv,len=?
The system responds:    LEN=a-b-c-d-e-f
                          a is an SM number between 1 and 192
                          b is a number between 0 and 7
                          c is a number between 0 and 9
                          d is 0 or 1
```

```
         e is a number between 0 and 7
         f is a number between 0 and 3

     append text, hit ? for more info.,
     hit (CR) to execute, or hit (DEL) to reject
     OP:CONV,LEN=
```

The system lists the possible entries for the values ('a' through 'f') of parameter LEN of this message. The underline character represents the position of the cursor on the screen.

2.6.3 Getting Help for an Error Message

If you make a mistake when you enter an IM, the system responds with an error message. You can invoke the help facility after receiving such a message and get more information about how the message should be composed.

Here is an example of getting help after the system sends an error message:

```
EXAMPLE 3
If you enter:          op:hdwchk,clink;
The system responds:
                       ?D - unexpected keyword CLINK
                       enter a new command OR ? for more information

          If you now type a question mark, you will obtain the following system response:
                       unexpected keyword CLINK
                       The input matches 3 messages.

                       (Further errors maybe revealed after
                        a single message has been selected.)

                       1)  OP:HDWCHK,a;
                               a is MMP=b-c, CMP=b-d, FPC=b, PPC=b,
                               ONTC=b, MSCU=b, or CLNK=e-f-g-h;
                               must choose one

                       2)  OP:HDWCHK,CM;

                       3)  OP:HDWCHK,NCREF,a;
                               a is PRIM=b, SEC=b, XC=b; REF1=b, REF2=b,
                               REF3=b, REF4=b, REF5=b, REF6=b, REF7=b,
                               REF8=b; must choose one

                       enter a new command OR ? for more information
```

Typing the question mark after an error message has the same effect as typing a message fragment followed by a question mark. The same kind of help is provided. In this case, three messages begin with OP:HDWCHK. The unexpected keyword error is ignored.

2.6.4 The Second Level of Help: Prompting

Once you have gotten help as described in the previous section, you can get an additional kind of help: prompting. If you type a question mark again after receiving the help described in Examples 1, 2, and 3, you are prompted for each entry in the message until it is complete.

If you type a question mark in response to Example 2, you are prompted for each parameter value (a through f) for parameter LEN. In Examples 1 and 3, you are asked to choose one message from the list that is displayed. Then

you are prompted to complete the chosen message.

2.6.5 Ending the Help Session

You may exit the help facility at any time by pressing the "Delete" key. If you are being prompted, then you must press the "Delete" key once to end the prompting, and press it a second time to end the help session.

2.7 INPUT MESSAGE EDIT AND HISTORY

The input message edit and history function is designed to assist the user in retrieving and editing input messages already entered on a given terminal. It provides the following capabilities to the user:

- A record of input messages entered during a session.

- Retrieval by history buffer line number, by string, or by last input message entered.

- Edit of retrieved input messages.

2.7.1 Availability

The input message edit and history function is provided on the following terminal types:

- Maintenance Control Center (MCC)

- Supplementary Trunk and Line Workstation (STLWS)

- Switching Control Center (SCC)

- Recent Change and Verify (RC/V)

2.7.2 Related Input/Output Messages

The input message edit and history function includes four input messages and an output message. The input messages apply only to the terminal upon which they are entered. See the appropriate manual pages for more information.

IM Message ID	Purpose
ALW:HIST	allow history recording
INH:HIST	inhibit history recording
CLR:HIST	clear history buffer
OP:HIST	displays entries from the history buffer

OM Message ID	Purpose
OP:HIST	response to the OP:HIST input message

2.7.3 History

Each terminal keeps a separate history buffer containing only the input messages entered at that terminal. Input messages are saved only when history recording is allowed. By default, history recording is allowed when a terminal comes into service.

If history recording is allowed and the input message is not a help input message, then the submitted input message is saved in the history buffer.

Function associated input messages that access help information and retrieve or edit input messages are not stored, regardless of the status of history recording.

2.7.3.1 Service Affecting or Sensitive Input Messages

It is recommended that the user inhibit history recording before entering any input messages that are sensitive or may cause interruptions or degradations of service. This prevents the service affecting input message from being accidentally retrieved and re-executed. After the sensitive or service affecting input message is executed, history recording can be allowed.

2.7.3.2 History Buffer Line Numbers

The history buffer can have a maximum of 200 entries at any given time. The buffer line numbers, however, begin at 1 and have a maximum range of 999.

Numbering of the buffer lines continues to 1000 even though only the last 200 input messages are in the buffer at any given time. The input message assigned to buffer line number 1000 triggers the following sequence:

- It is reassigned to line number 200.

- The last 199 entries already in the buffer are renumbered 1 through 199.

- The incrementation of buffer line numbers resumes with 201.

2.7.3.3 Clearing the History Buffer

It is recommended that the user clear out the history buffer at the end of each terminal session. This helps keep users from accessing input messages and corrupting the switch database by accident or intent. The buffer is always empty when a terminal comes into service.

2.7.4 Special Characters

The characters '#' and '^' are used for line editing control. The '#' character indicates that a retrieval process is to commence. The '^' indicates that a string follows. The string may be null.

2.7.5 Input Message Retrieval Procedure

A input message can be retrieved by one of the following methods:

Code	Explanation
#*n*>	Retrieve by history buffer line number. This retrieves line number 'n' from the history buffer.
##	Retrieve last input message. This is a shorthand method to retrieve the most recently stored input message from the history buffer.
#^a	Retrieve by string. This retrieves the most recently stored input message that contains the string 'a' from the history buffer.

2.7.6 Command Modes

The modes associated with the input message edit and history function are:

- command entry mode

- edit mode

2.7.6.1 Command Entry mode

Command entry mode is the default mode. Input messages entered in command entry mode are stored in the

18

history buffer, if history recording is allowed. The prompt for command entry mode is '<'.

Enter '?' to request help information.

2.7.6.2 Edit Mode

From command entry mode, edit mode is accessed by entering one of the special character sequences that indicate history buffer retrieval is being requested.

If the indicated buffer line number or string is found in the history buffer, the retrieved input message becomes the "pending" input message and the craftshell switches to edit mode. If an available input message is not identified, a "Command Unavailable" error message is output and the craftshell remains in command entry mode.

The user stays in edit mode until he/she either submits the pending input message for execution or cancels the edit session. After each edit, the latest version of the pending input message is displayed, along with the edit mode prompt (<<). At the edit mode prompt, the user can perform any of the following actions:

- enter '?' to request input message edit help information.

- enter (cr) (carriage return) to execute the pending input message.

- enter (del) (delete) to return to command entry mode, without executing the pending input message.

- retrieve another saved input message by any of the available methods.

- continue editing the pending input message.

- enter an input message.

To stay in edit mode, a pending input message must always be available. If, after an edit cycle is complete, a pending input message is not available, the craftshell returns to command entry mode.

If a carriage return or semi-colon (;) is entered at the edit prompt (<<), the pending input message is submitted to the craftshell for execution and:

- the edit session is terminated

- the craftshell returns to command entry mode

- the command entry mode prompt (<) is displayed.

If the input message entered is incorrect, an appropriate error message is displayed by the craftshell.

2.7.7 Help

If a lone question mark (?) is entered at the edit prompt (<<), a brief input message edit help message is displayed, followed by the latest version of the pending input message and the edit mode prompt (<<).

If the user enters an edited input message with invalid syntax, the input message edit help information is displayed after the "Command Not Recognized" error message.

2.7.8 Line Editing

Input message editing provides the capability to retrieve saved input messages and re-execute them. Retrieved input messages may be edited using substitutions, appends, or a combination of both before re-execution.

2.7.8.1 Editing Definitions

Append   = A string is added to the end of the pending message.

Substitute   = A substitution is performed only at the right-most occurrence of the specified *oldstring* within the pending input message. This permits the user to easily update just the last characters(s) of an input string.

Nuances   = A null *oldstring* causes the replacement string to be appended to the end of the pending input message. A null *newstring* causes the *oldstring* to be deleted from the pending input message.

2.7.8.2 Retrieval/Editing Options

Code	Explanation
^*newstring*	The `newstring` text is appended to the end of the pending input message or the last input message entered.
^*oldstring*^*newstring*	Substitute the new string for the old string to the pending input message or the last input message entered.
#*n*^*newstring*	Retrieve line number from the history buffer (#*n*) and append the *newstring* to it.
##^*newstring*	Retrieve the most recently stored input message from the history buffer (##) and append the *newstring* to it.
#^*searchstring*^*newstring*	Retrieve the most recently stored input message containing the *searchstring* from the history buffer (#^*searchstring*) and append the *newstring* to it.
#*n*^*oldstring*^*newstring*	Retrieve line number from the history buffer (#*n*) and substitutes the last occurrence of the *oldstring* with the *newstring*.
##^*oldstring*^*newstring*	Retrieve the most recently stored input message from the history buffer (##) and substitute the last occurrence of the *oldstring* with the *newstring*.
#^*searchstring*^*oldstring*^*newstring*	Retrieve the most recently stored input message containing the *searchstring* from the history buffer (#^*searchstring*) and substitute the last occurrence of the *oldstring* with *newstring*.

2.7.9 Error Cases

The input message edit and history function provides specific error messages to assist the user when problems are encountered when retrieving or editing an input message and when invalid conditions for an OP:HIST input message exist.

For error conditions that occur during the retrieve or edit procedures, the following results are possible:

Result	Explanation
Remain in command entry mode.	If an attempt to retrieve a input message (and go into edit mode) from *command entry* mode fails, there is no pending input message available. An error message is output, followed by the command entry mode prompt (<).
Remain in edit mode.	If there is a pending input message available, subsequent error conditions generates an error message followed by the pending input message and the edit mode prompt (<<). The pending input message is the result of the last successful retrieve or edit.
Go to edit mode.	If a combined retrieve and edit input message is entered at the command entry mode prompt (<), it may succeed for the retrieve and fail for the edit. An error message is output (for the failed edit), followed by the pending input message (the successful retrieve), and the edit mode prompt (<<).
Go to command entry mode.	If a successful edit results in a null pending input message, the command entry mode prompt (<) is output. No error message is output, since the edit was successful.

2.8 MCC DISPLAY PAGES

Master Control Center (MCC) display pages are used to perform maintenance and administrative functions for the *5ESS*® switch.

See document 235-105-110 System Maintenance Requirements and Tools for a further explanation of MCC display pages.

2.9 *5ESS*® SWITCH DIRECTORY STRUCTURE AND PATHNAMES

A *5ESS*® switch file system is a set of directories, ordinary files, and special device files.

- A directory is a collection of names of files and other directories.

- An ordinary file (also called a `flat file') is a collection of ASCII text. It may consist of ASCII characters, divided into lines demarcated by the new-line character, or it may consist of binary code that will be copied into main memory when a program is executed.

- A special device file supports input/output activity on a physical device, such as a terminal, magnetic tape, or disk partition. A request to read or write a special device file results in activation of the associated device. However, there is not necessarily a one-to-one correspondence between physical devices in the system and special device files. A single device may have more than one special device file. For example, four different special device files are used to read the same magnetic tape with high or low density and with or without rewind. Most special device files are located in a directory named /dev.

The set of directories and files that comprise a file system is organized into a hierarchical tree structure, illustrated in Figure 2-1 . In this example:

- the database, dev, etc, lib, tmp, and cft directories all descend from the "root" directory

- the database directory contains a number of ordinary files, including the ecd and appecd databases

- the dev directory contains a number of special device files

- the cft directory contains various subdirectories, which in turn contain ordinary files and additional subdirectories
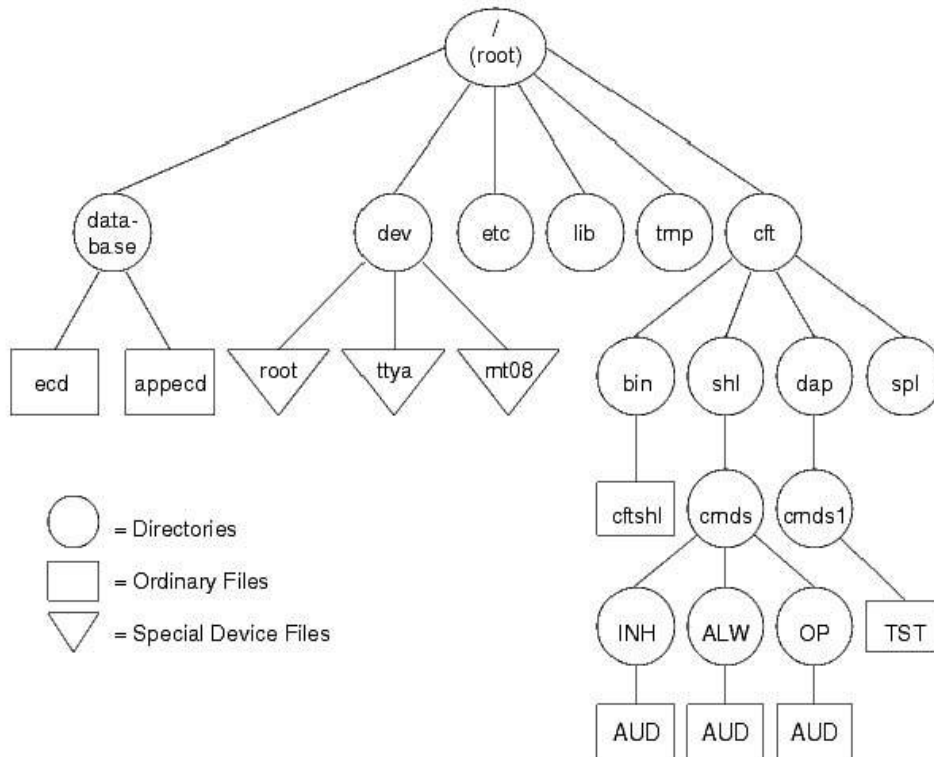
Figure 2-1 File System Structure

In the 5ESS® switch system, the database, etc, and tmp directories are themselves the "root" directories of separate file systems that are mounted on the root file system during a system bootstrap. These four file systems (root, database, etc, and tmp) constitute the minimum set of file systems that are required for the 5ESS® switch system to function. Backup copies of these file systems may also be provided, in case one of the primary file systems becomes damaged in a way that would prevent the system from operating successfully. If the system is bootstrapped on the backup file systems, the names of the directories where they are mounted remains the same. Other file systems may also be mounted automatically during a bootstrap, or they may be mounted manually by using ALW:FSYS-MOUNT.

Every file and directory in the system is identified by a unique pathname that indicates its location in the hierarchical file system structure. Many 5ESS® switch IMs require the specification of a pathname. There are two types of pathnames that may be specified:

- Full pathnames begin with a slash (/). This is the name of the "root" directory of the root file system, and is followed by the name of the directory that is immediately beneath the "root" directory, followed by another slash, then the next directory name, and so forth, until the name of the specified file or directory is reached.

  For example: `FN="/user/a*/smo/test"`

- Relative pathnames begin with the name of a directory or file at an arbitrary point in the hierarchy, with no preceding slash. Their specification is relative to the immediately preceding directory. This is sometimes referred to as the base directory.

Because they contain lower-case alphabetic characters and slashes, pathnames must always be surrounded by quotation marks ("") when entered in an IM.

2.10 BINARY-DECIMAL-HEXADECIMAL CONVERSIONS

Both input and output messages use values in hexadecimal (base 16), decimal (base 10), octal (base 8), and binary (base 2). To convert binary numbers to hexadecimal, do the following:

- Separate the binary number into groups of four, starting with the rightmost number.

- Convert each group of four into its hexadecimal equivalent. (See Table 2-5 of the User Guidelines.)

- Example:

```
001000011110        Binary number
0010 0001 1110      Binary number in groups of four
2    1    E         Hexadecimal equivalent
```

To convert hexadecimal numbers to binary, perform the above procedure in reverse order.

To convert numbers from hexadecimal to decimal or vice-versa, refer to Table 2-4 of the User Guidelines. The left column contains hexadecimal numbers without the units digits. The units digits are shown in the first row.

Table 2-1   Craftshell Responses

Response	Meaning
?D - incorrect value 'r' - arg #s of t	The craftshell encountered an IM with an incorrect parameter value. r = The incorrect parameter value. s = The index number of the parameter value. t = The corresponding parameter name.
?D - keyword a typed out of order	The craftshell encountered an IM with the order of parameter names from two parameter blocks reversed. a = The first out-of-order parameter name that the craftshell encountered.
?D - missing keyword	The craftshell encountered an IM with a missing parameter name.
?D - unexpected keyword b	The craftshell encountered an IM with a parameter name that it does not recognize. b = The parameter name that the craftshell does not recognize.
?E BAD IM CATALOG	The craftshell detected a problem with the IM catalog. The input request may or may not succeed. The catalog may be missing or corrupted, or UPD:IMCAT may not have been entered after a new catalog was installed. Refer to

	REPT:CFTSHL output message and UPD:IMCAT IM for further information.
`?E CFTSHL HANGUP RECEIVED`	The craftshell terminated because it received a hang-up signal.
`?E CFTSHL TERMINAL READING ERROR, ERRNO=c`	The craftshell encountered an error while reading input from a terminal. c = System error code (refer to the System Error Codes Appendix, APP:SYSERR, in the Output Messages manual).
`?E CFTSHL UNABLE TO FORK PROCESS d`	The craftshell can not fork a new process to execute a command. d = Full pathname of the process that was not executed because of the fork failure.
`?E COMMAND NOT RECOGNIZED`	This Line Edit and History error can occur when a string (recall, append or substitute string) was not prepended with a caret (^), a recall command was entered that did not properly specify which stored command was being recalled or there were too many 'caret' characters in an edit command.
`?E - input matches multiple messages`	The typed input request matches more than one input message in the IM catalog and the craftshell can not determine which message to execute. To obtain further information enter the help facility by terminating the input request with a question mark ('?'). If the desired input request can not be satisfied after accessing the help facility, refer to the TECHNICAL ASSISTANCE portion of the INTRODUCTION section of the Output Messages manual.
`?E NO COMMANDS ARE STORED`	A Line Edit and History recall command was entered when there are no commands in the history buffer to recall. The No Commands Are Stored error takes precedence over the other Line Edit error messages.
`?E STRING NOT FOUND`	This Line Edit and History error can occur for two reasons: A substitute command was entered where the 'OLD' string is not present in the pending command. A recall by string command was entered where the recall string is not present in any of the commands stored in the command history buffer.
`?E STRING TOO LONG`	This Line Edit and History error occurs when an append or substitute command is entered that would result in a command longer than the legal limit of 256 characters. The append or substitute is rejected.
`?E - syntax error; invalid e`	The craftshell encountered an IM syntax error. e = The part of the IM that the craftshell identified as invalid.
`?E - syntax error; unexpected 'f' after g`	The craftshell encountered an IM syntax error. f = First invalid part of the IM. g = Last valid part of the IM.
`?E - terminator must be 'q'`	The craftshell encountered an improper IM terminator. q = Proper IM terminator character.
`?E UNABLE TO INITIATE COMMAND h`	The craftshell is unable to execute an input request because it

24

	encountered a zero-length file.
	h = Full pathname of the command that cannot be initiated.
`?E UNABLE TO MATCH COMMAND i`	An input command was entered that does not exist in the craftshell search directories.
	i = Full pathname of the non-existent command code
`?E UNAVAILABLE COMMAND, CMD = j TO k`	A Line Edit and History recall command was entered that identified a command that is out of the available range. To assist the user, the available range is printed as part of the error message.
	j = Lowest command number of available range
	k = Highest command number of available range
`?E INVALID COMMAND`	The craftshell is unable to execute an input request because it encountered a non-executable file or because it located a command directory but no executable file.
`?V - invalid command code l`	The craftshell encountered an IM with an invalid command code.
	l = The command code that the craftshell identified as invalid.
`NG - FEATURE NOT AVAILABLE`	The requested action failed because the feature required to process the request is not present in the switch or the given module.
`NG - FILE OPEN FAILURE`	An OP:HIST input command was entered but the output file could not be opened to print the information.
`NG - NO COMMANDS ARE STORED`	An OP:HIST input command was entered when there are no commands in the history buffer to output.
`NG - RANGE UNAVAILABLE, CMD = m TO n`	An OP:HIST input command was entered that specified a range (either explicitly or implicitly) that does not match any available stored command. To assist the user, the available range is printed as part of the error message.
	m = Lowest command number of available range
	n = Highest command number of available range
`NG - SM o is isolated`	The craftshell encountered an input request targeted for a switching module (SM) that is in the isolated state.
	o = Processor number of the isolated SM.
`NG - SM p is not equipped`	The craftshell encountered an input request targeted for an SM that is not equipped.
	p = Processor number of the unequipped SM.
`NG - SYNTAX ERROR IN PREVIOUS MESSAGE`	The EXC:PREV command was entered to execute a command that has a syntax error.
`RL - UNABLE TO EXECUTE COMMAND`	The craftshell is unable to communicate with another process to execute an input request. Retry request later.
`RL - UNABLE TO SERVICE REQUEST`	The craftshell is unable to execute an input request due to unavailable system resources such as system overload, excessive queue lengths, or a busy process. Retry request

235-600-700   December 2003

later.

Table 2-2   Summary of Parts of a Message Format

Term or Item	Definition
Block of Parameters	Any field but first field of message. Contains parameter names and parameter values.
Braces { }	Enclose two or more entries, separated by OR bars, of which the user must input one, but only one. Not input as part of the message.
Brackets [ ]	Enclose optional entries. Not input as part of the message.
Command Code	First field of message. Contains command code.
Data Field	Optional blocks of parameters in messages.
Equal Sign =	Separates parameter name and parameter value.
Exclamation Point !	Message terminating character that does one of the following: • In certain messages, causes the command to be executed at an appropriate break point. • Causes the system to expect more data field parameters to follow on the next line of input. • Causes the entered portion of the command to be executed as an immediate action (any ordinary message). Also causes a special colon (:) prompt to be issued. The terminal will not print output messages until the normal semicolon (;) terminating character is entered.
Field	Basic structural unit of message. Variable length. Separated by colons. There are at least two fields per message: command code and parameter block(s), in that order.
Keyword	The term keyword is often used in error messages to refer to a parameter name.
Message Length	Number of characters including spaces within the message (up to 255 characters). This may exceed one line.
Number Base	Default is decimal. Otherwise, numbers input must be preceded by B' (binary), O' (octal), or H' (hexadecimal).
OR bar		Separates entries within braces or brackets. One entry, but only one, must be selected. If selections are enclosed by braces, one entry must be input; if enclosed by brackets, the entry is optional. Not input as part of the message.
Parameter	Parameter name and its associated parameter values, if any. Two or more parameters within a block of parameters are separated by commas.
Parameter Name	Mnemonic identifier of a unit, function, or condition. May have parameter values associated with it. Error messages often refer to parameter names as keywords. Input as shown.
Parameter Value	Separated from parameter name by equal sign (=). Often represented by a variable in message format. Can be input as a single entry, as multiple entries in the form a-b-c, or as a range in the form x&&y.
Pathname	Starts with a slash (/), with every file and directory name separated by a slash. When entered, must be immediately preceded and followed by quotation marks (").
Question Mark ?	Help request character. A question mark may follow a partial message or previously entered message that was erroneous. Typing ? after a help guidance message will cause it to enter prompting mode.
Quotation Marks ""	Enclose pathname of file, directory, or device. Also enclose any text strings entered. Input as shown.
Semicolon ;	Message terminating character that causes the command to be executed as an immediate action.
&&	Indicates a range in messages.
Variable	Shown as a lowercase letter in manual pages. Replace with user-supplied information.
Verb	Known as the command code. Shows what type of action is to be taken. May be an abbreviation of the command code.

Table 2-3   Summary of Pathnames

Pathname	Meaning
/	Full pathname of the "root" directory of the *root* file system.
/data base	Full pathname of the *data base* directory which is located directly under /.
/cft/shl/cmds	Full pathname of the *cmds* directory, which is located under the *shl* directory, which is located under the *cft* directory, which is located directly under /.
/cft/shl/cmds/INH/AUD	Full pathname of the *AUD* program which is executed by the INH:AUD message.
INH/AUD	Relative pathname of the same *AUD* program, relative to the */cft/shl/cmds* base directory.
OP/AUD	Relative pathname of the *AUD* program which is executed by the OP:AUD message, relative to the */cft/shl/cmds* base directory.
/dev/root	Full pathname of the special device file for the disk partition that contains the *root* file system (mounted as /).
/dev/broot	Full pathname of the special device file for the disk partition that contains the backup copy of the *root* file system (also mounted as /).
/dev/db	Full pathname of the special device file for the disk partition that contains the *db* file system (mounted as */data base*).
/dev/bdb	Full pathname of the special device file for the disk partition that contains the backup copy of the *db* file system (also mounted as */data base*).
/dev/mt00	Full pathname of the high density magnetic tape device that is attached to input/output processor (IOP) number 0 with an option 0 (rewinds a tape after reading from it or writing to it).
/dev/mt04	Full pathname of the high density magnetic tape device that is attached to IOP number 0 with an option 4 (does not rewind a tape after reading from it or writing to it).
/dev/mt08	Full pathname of the low density magnetic tape device that is attached to IOP number 0 with an option 8 (rewinds a tape after reading from it or writing to it).
/dev/mt0c	Full pathname of the low density magnetic tape device that is attached to IOP number 0 with an option c (does not rewind a tape after reading from it or writing to it).
/dev/mt10	Full pathname of the high density magnetic tape device that is attached to IOP number 1 with an option 0 (rewinds a tape after reading from it or writing to it).
/dev/mt14	Full pathname of the high density magnetic tape device that is attached to IOP number 1 with an option 4 (does not rewind a tape after reading from it or writing to it).
/dev/mt18	Full pathname of the low density magnetic tape device that is attached to IOP number 1 with an option 9 (rewinds a tape after reading from it or writing to it).
/dev/mt1c	Full pathname of the low density magnetic tape device that is attached to IOP number 1 with an option c (does not rewind a tape after reading from it or writing to it).

Table 2-4   Binary-Octal-Decimal-Hexadecimal Conversions

Binary	Octal	Decimal	Hexadecimal
0000	00	00	0
0001	01	01	1
0010	02	02	2
0011	03	03	3
0100	04	04	4
0101	05	05	5
0110	06	06	6
0111	07	07	7
1000	10	08	8
1001	11	09	9
1010	12	10	A
1011	13	11	B
1100	14	12	C
1101	15	13	D
1110	16	14	E
1111	17	15	F

Table 2-5   Hexadecimal-Decimal Conversion

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	000	001	002	003	004	005	006	007	008	009	010	011	012	013	014	015
1	016	017	018	019	020	021	022	023	024	025	026	027	028	029	030	031
2	032	033	034	035	036	037	038	039	040	041	042	043	044	045	046	047
3	048	049	050	051	052	053	054	055	056	057	058	059	060	061	062	063
4	064	065	066	067	068	069	070	071	072	073	074	075	076	077	078	079
5	080	081	082	083	084	085	086	087	088	089	090	091	092	093	094	095
6	096	097	098	099	100	101	102	103	104	105	106	107	108	109	110	111
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127

8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255

Frequency–Agile FM Surveillance Bug

Overview

This is a simple little project to turn a "car MP3 player" into a frequency–agile audio surveillance bug.

The car MP3 player used for this project was listed as "Portable Car FM Transmitter MP3 Player SD TF USB LCD Display Small sports Auto" on eBay and appears to be sold by various users for around $4 each. Like most of these Chinese toys, there is no documentation or even manufacture info. Oh, and no FCC ID either...

These MP3 players have an internal FM transmitter based on the Beken BK1085, which is in a 16–pin SSOP package with pins 7 and 8 appearing to be the left/right analog audio inputs. I wasn't able to find a datasheet for this chip, so that was mostly determined by poking around a bit with an oscilloscope and signal generator.

The MP3 player itself is designed to plug into a vehicle's cigarette lighter (+12 VDC). It will transmit audio from either an USB connection or SD card to the vehicle's FM radio via the internal transmitter. The transmitter tunes over the standard FM broadcast frequency range of 87.5 MHz to 108 MHz in 100 kHz steps. The RF output power is around 2 milliwatts (+3 dBm). The stock antenna is just a short wire coupled to the +12 VDC input line via a 0.01 µF capacitor.

On the DC input power cable, the red wire is +5 VDC, the green wire is ground, and the copper–colored wire is the antenna. There is a 78M05 regulator on the +12 VDC input to supply the circuit with clean +5 VDC. The current draw is around 45 milliamps.

The LCD displays the current transmitting frequency and has a white LED backlight, both of these may be removed to reduce power consumption and to save space.

To turn the device into a surveillance bug, an electret microphone and a small audio amplifier will be used to directly drive one of the analog audio input pins on the Beken BK1085.
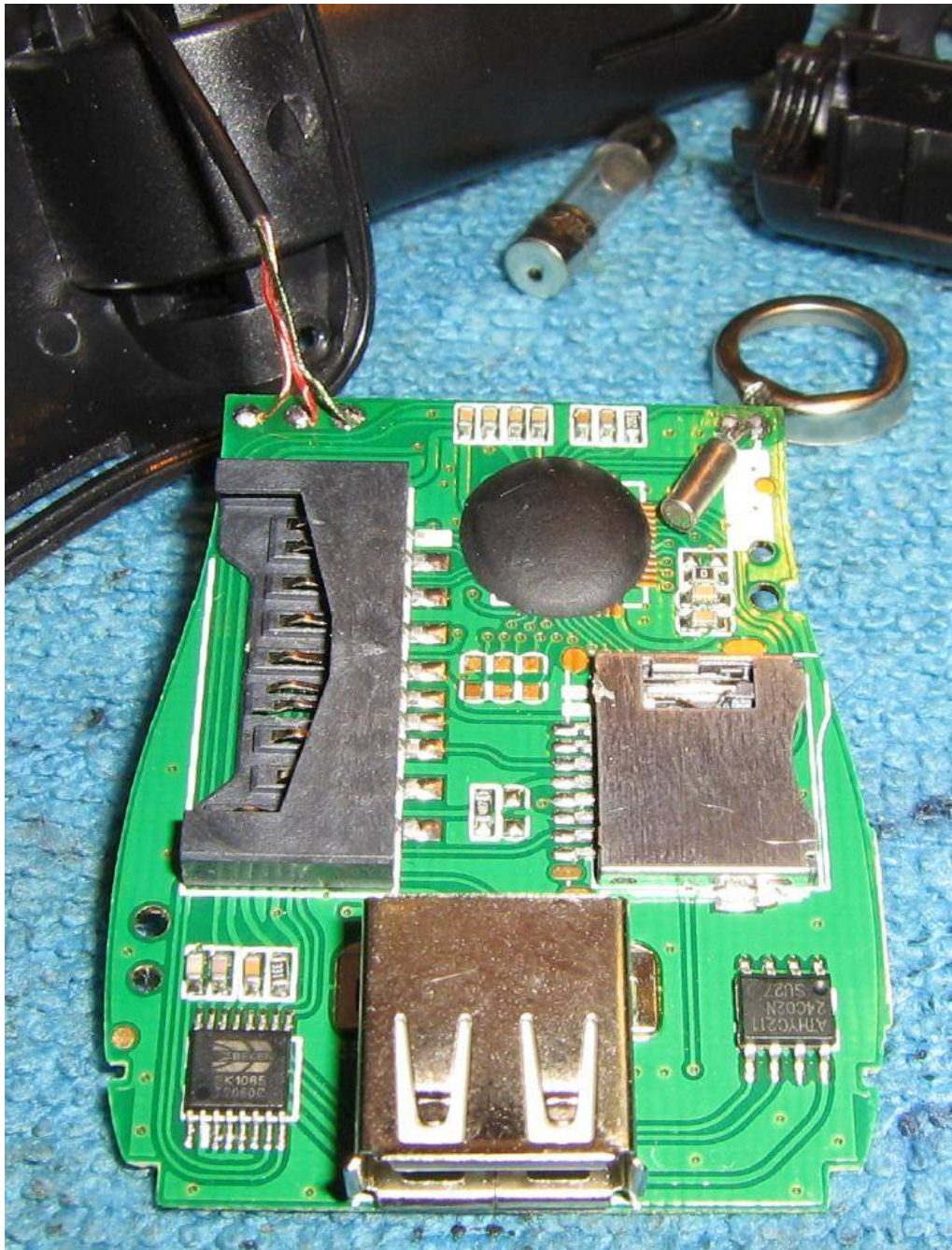
Pictures & Construction Notes

Internal overview.

The 78M05 voltage regulator board is off to the left.

The red and black wires going to the LCD display are for its backlight.
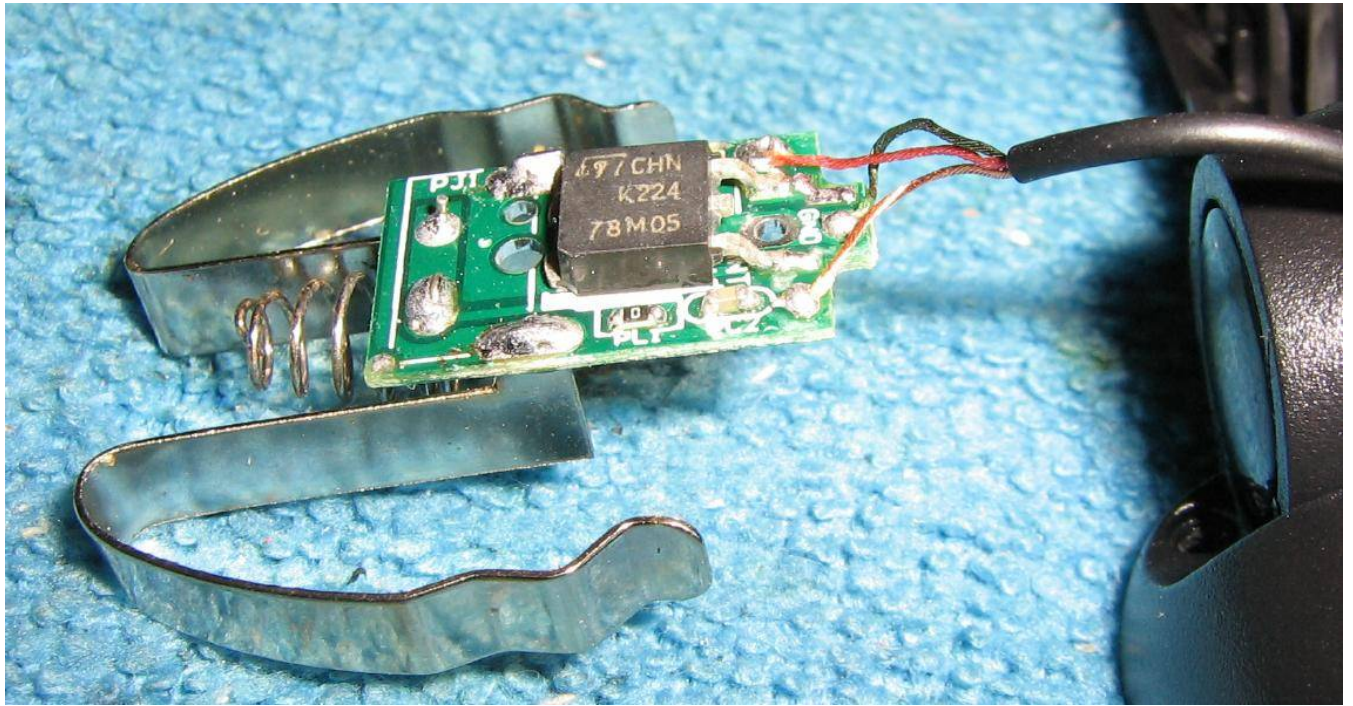
The little black rectangle below the LCD display is the infrared sensor for the accompanying remote control.

Closeup of the rear of the MP3 player showing the different connectors for the digital file/audio inputs.

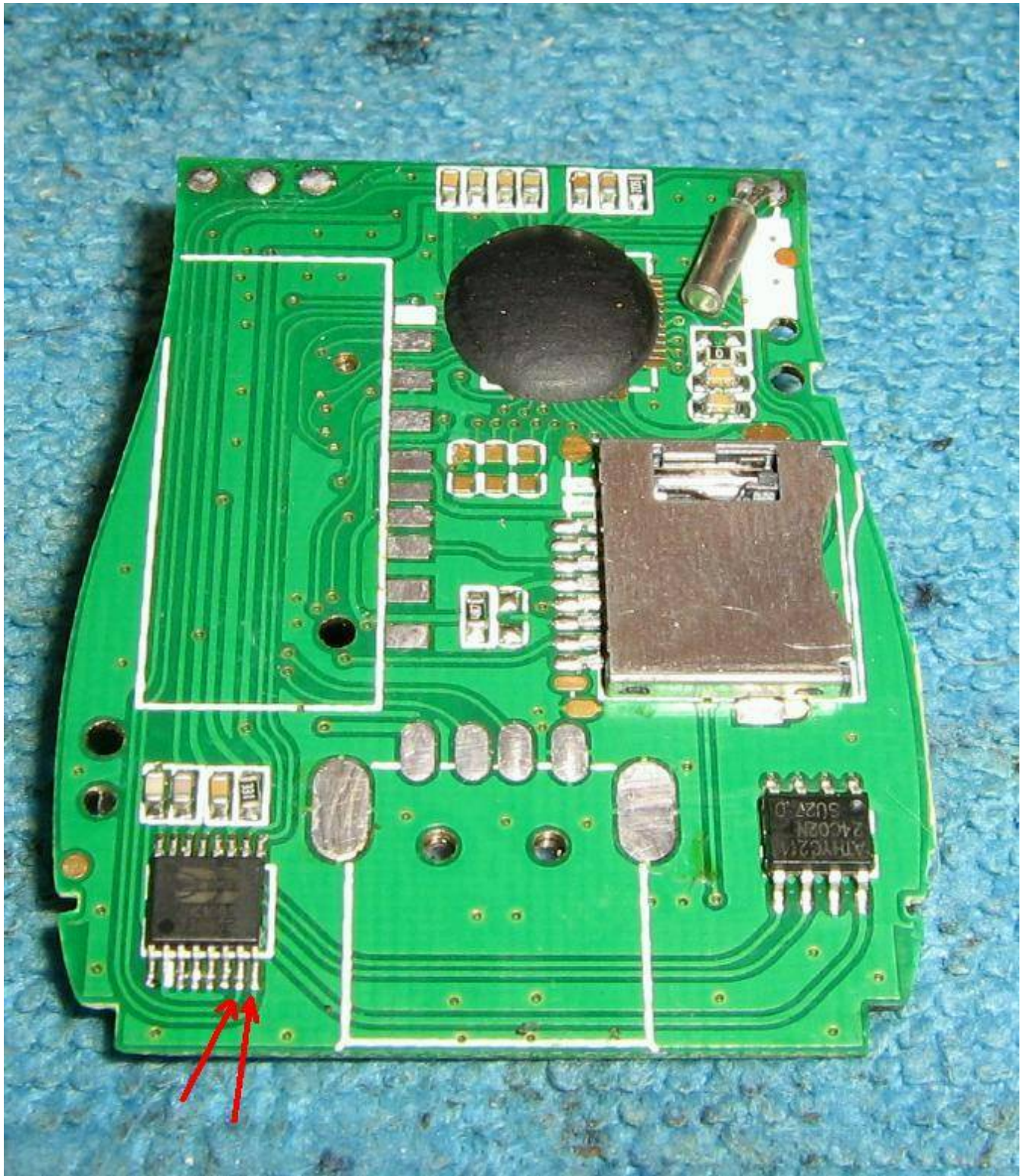These can be removed to save space.

The Beken BK1085 FM transmitter is the 16−pin IC on the lower−left.

Closeup of the 78M05 voltage regulator board.

The red wire is the +5 VDC output.

The green wire is ground.

The copper−colored wire is the antenna which is AC−coupled to the incoming +12 VDC line via a 0.1 μF capacitor.

Removing some of the unnecessary connectors.

The analog audio inputs to the Beken BK1085 appear to be pins 7 and 8.
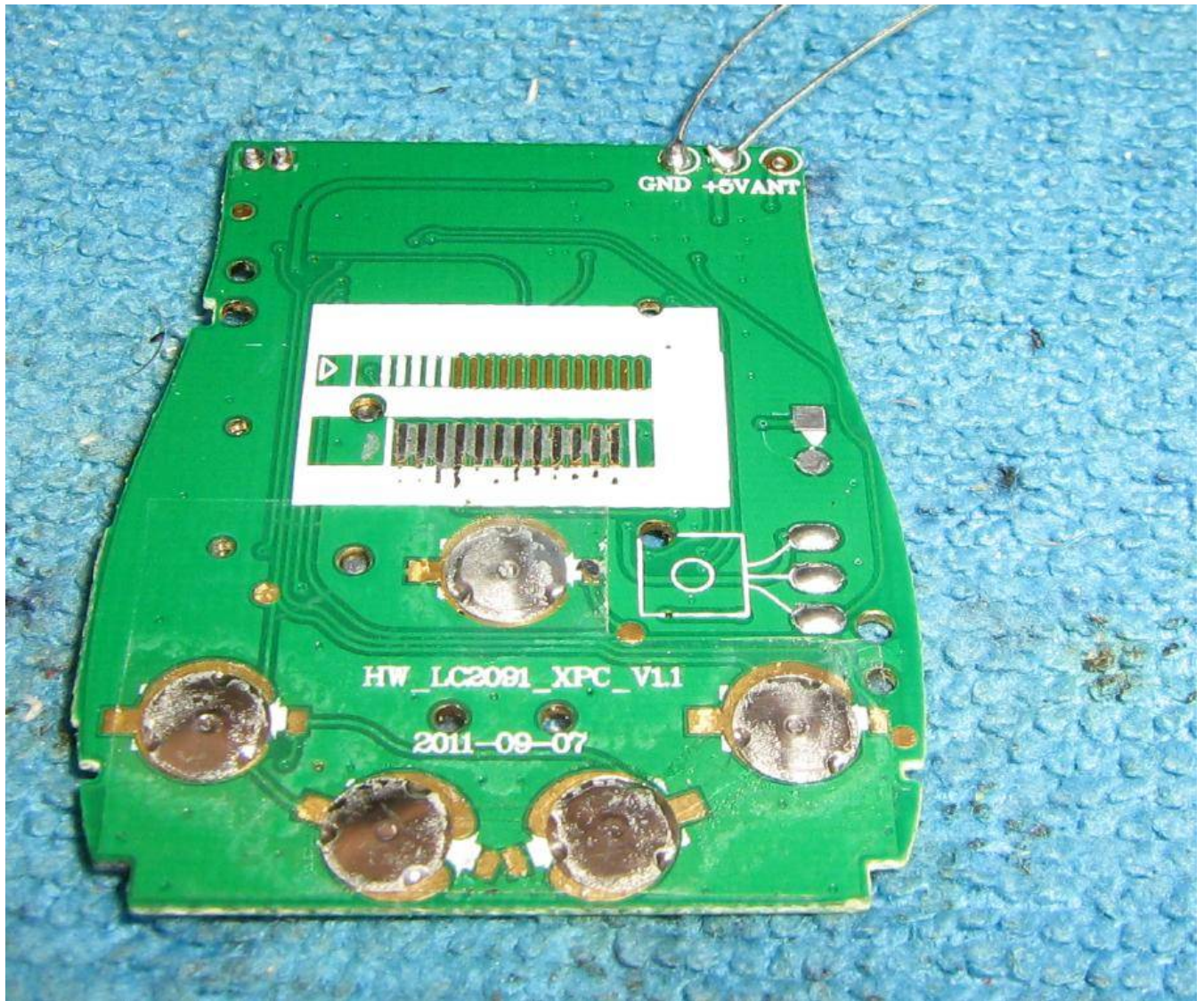
Removing the LCD display to save some space.

You'll need to double−check the output frequency using a frequency counter.

The two outside buttons along the bottom are "Frequency Down" (left) and "Frequency Up" (right).

The three connections along the top are (from left−to−right): ground, +5 VDC input, and the antenna.
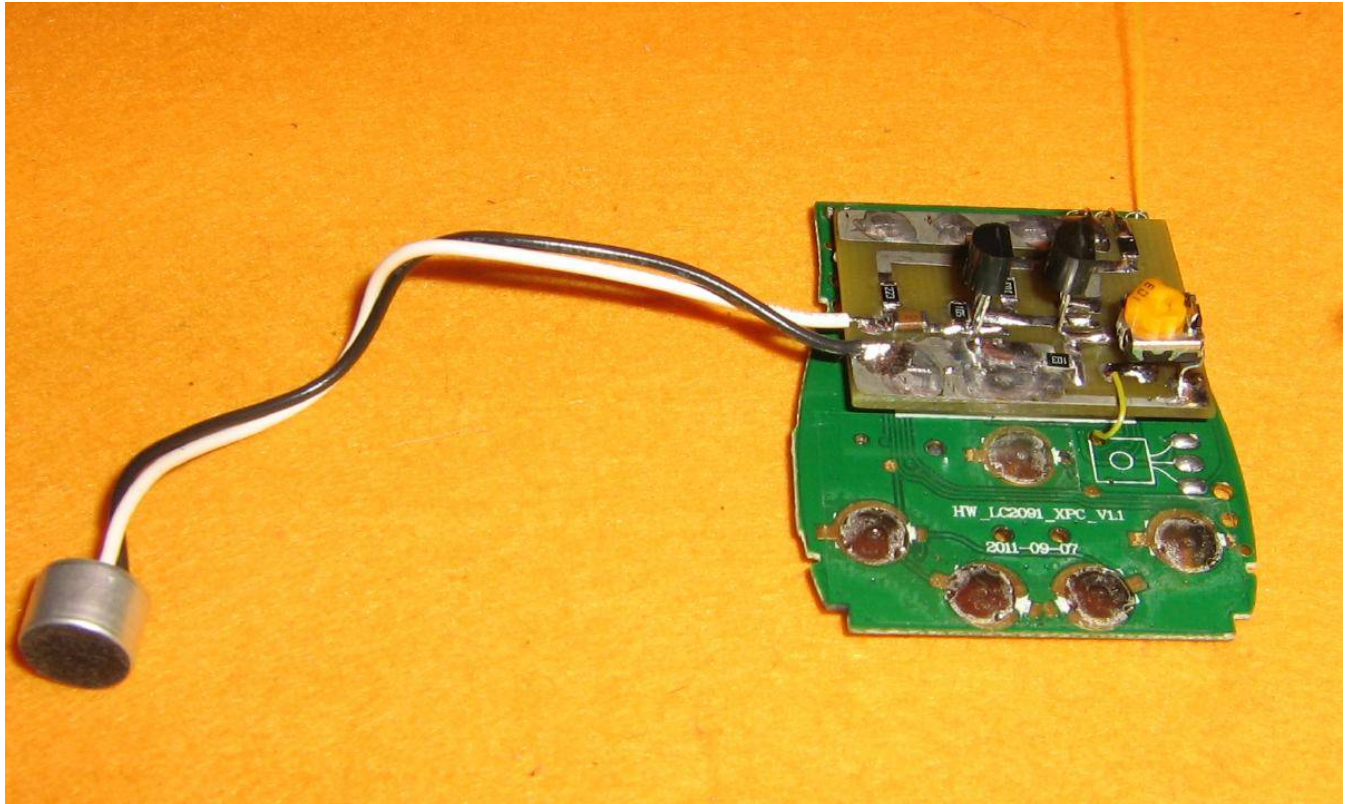
Adding a simple 2N5210−based transistor microphone amplifier circuit to the transmitter.

This was just an experimental circuit and using all surface−mount components would significantly reduce the size.

The antenna is just a 24−inch long piece of #30 gauge wire.

A Motorola MWA130, or similar MMIC amplifier, can be used to increase the RF output power. Note that this will further increase the drain on the batteries. It's usually more efficient to increase the overall gain of your receive station.

The audio inputs on the Beken BK1085 appear to be low−pass filtered, so ultrasonic subcarrier generators won't work well with this transmitter.
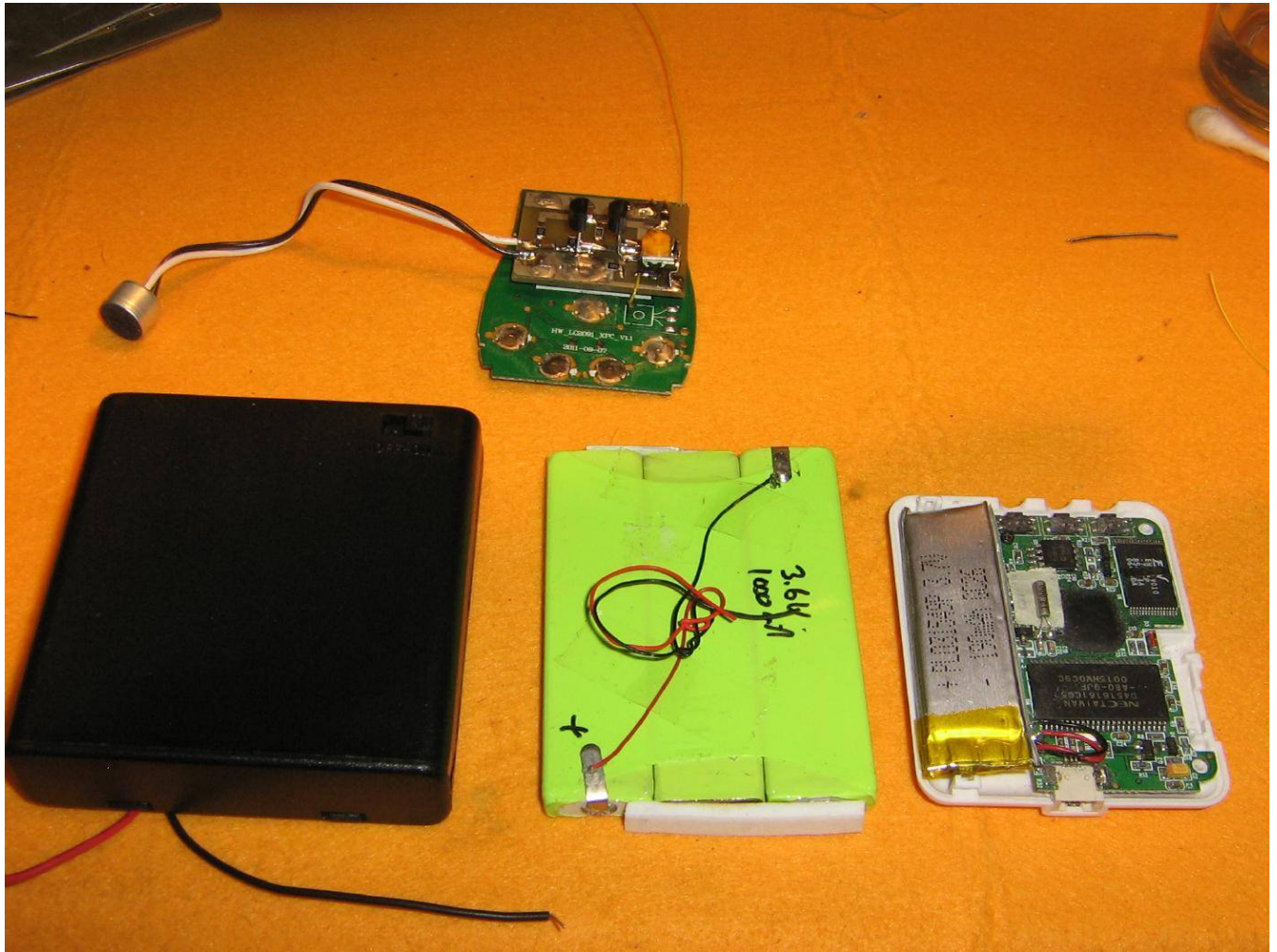
Example of some different DC power options.

The MP3 player and microphone amplifier are designed for +5 VDC, but will work down to around +3.5 VDC, with reduced RF power output.
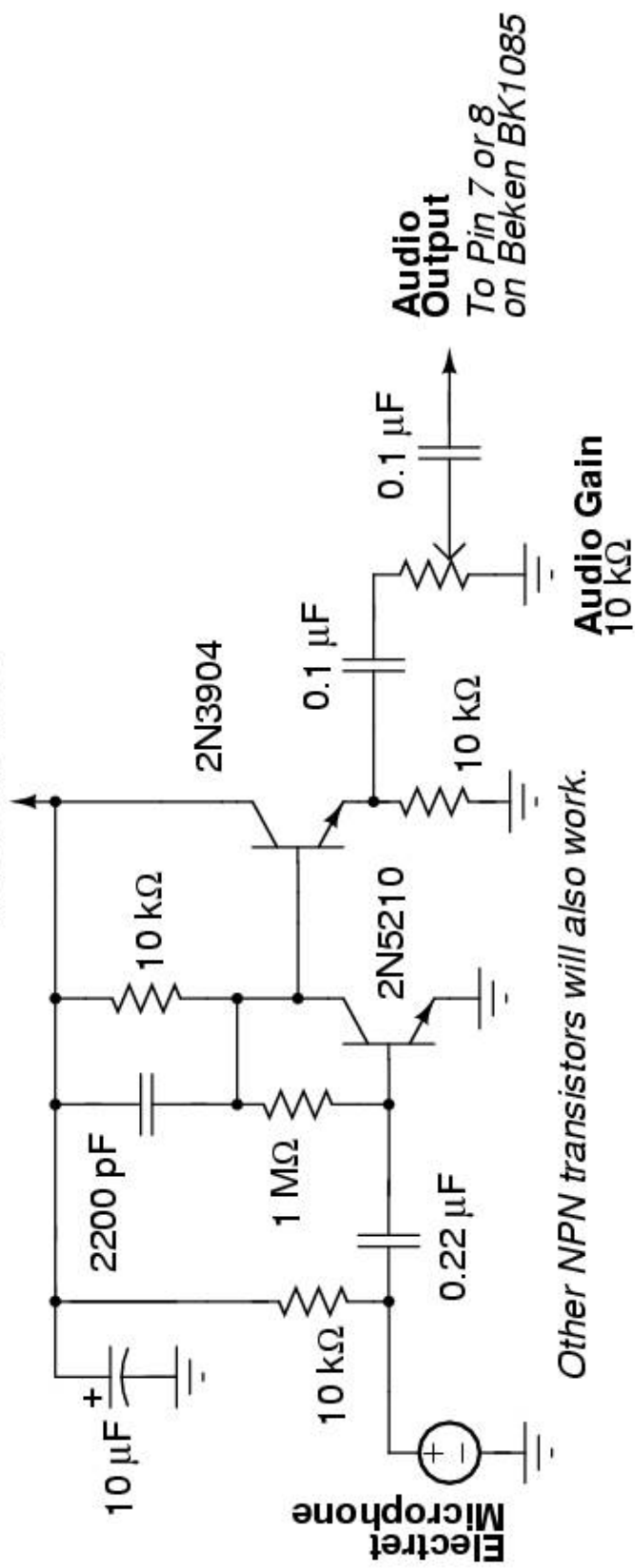
On the left, is a standard four "AA" battery holder from Radio Shack.

In the middle, is a salvaged +3.6 VDC, 1900 mA/h rechargable battery pack.

On the right, is a +3.6 VDC, 190 mA/h rechargable battery pack from one of those LCD "picture display" keychains. These can be recharged using an USB connection. You may want to try two of these in series.

Simple Elecret Microphone Amplifier

+3.5 to +5 VDC

2N3904

2N5210

10 kΩ

1 MΩ

2200 pF

10 μF

10 kΩ

0.22 μF

Electret Microphone

0.1 μF

10 kΩ

Audio Gain
10 kΩ

0.1 μF

Audio Output
To Pin 7 or 8 on Beken BK1085

Other NPN transistors will also work.

Subcarrier Generator for a FM Surveillance Bug

Overview

A subcarrier generator is a method commonly used for "masking" the audio from your surveillance bug by hiding it *within* the main RF carrier.

Normally, if you are using a conventional FM audio bug, anyone with a receiver tuned to your transmitting frequency will also be able to monitor your audio. This can easily give away your surveillance operation (or location), especially if you're using frequencies within the normal 87.5 MHz to 108 MHz FM broadcast band.

The idea of the subcarrier generator is to generate a "carrier on a carrier." In this example, we'll be using a frequency modulated 67 kHz subcarrier. A subcarrier of 67 kHz was choosen as this is often used for FCC–authorized Subsidiary Channel Authorization (SCA) transmissions within the FM broadcast band. This is often how Muzak, audio books, stock quotes, and other low–bandwidth services are transmitted over the airwaves. Decoders tuned for this specific 67 kHz subcarrier are quite easy to build, so we'll stick to using that frequency.

The subcarrier generator itself is based around a frequency–modulated oscillator using a common 555 timer IC. The 555 timer will be configured to generate a 67 kHz square wave output which is then frequency modulated by a signal from a LM833–based microphone pre–amplifier circuit.

The output from the 555 timer will then be low–pass filtered and attenuated before being applied to the *unfiltered* audio input on the FM surveillance bug. There must be no input low–pass filtering on the modulation input of your surveillance bug for this subcarrier method to work.

To receive the subcarrier–modulated audio, you need a receiver tuned to the proper carrier frequency and equipped with a subcarrier converter on the receiver's discriminator ("baseband") output. The discriminator output on the receiver is the directed demodulated audio before reaching any low–pass filtering or deemphasis stages.

In the Radio Shack PRO–2005/6 line of scanners, this output can be taken from the Samsung KA2243N wideband FM demodulator test point #1 (TP1 on the silkscreen). For other receivers, you may have to poke around a bit with an oscilloscope. Look for the 38 kHz stereo pilot tone when the receiver is tuned to a strong FM broadcast station.

You can make your own 67 kHz subcarrier converter by viewing the example circuit in the Signetics NE565 Phase Lock Loop datasheet. The NE565 can be difficult to track down, but they do appear on eBay from time–to–time.

Scanner	Wideband FM Discriminator Tap
Radio Shack PRO–2004	IC–1, pin 6 (TP3)
Radio Shack PRO–2005/6	IC–1, pin 10 (TP1)
Radio Shack PRO–2035/2042	IC–1, pin 10 (TP1)
Uniden BC–9000XLT	IC–9, pin 11
Icom R–1	IC–1, pin 9
AOR AR–8000	U3, pin 8

Pictures & Construction Notes
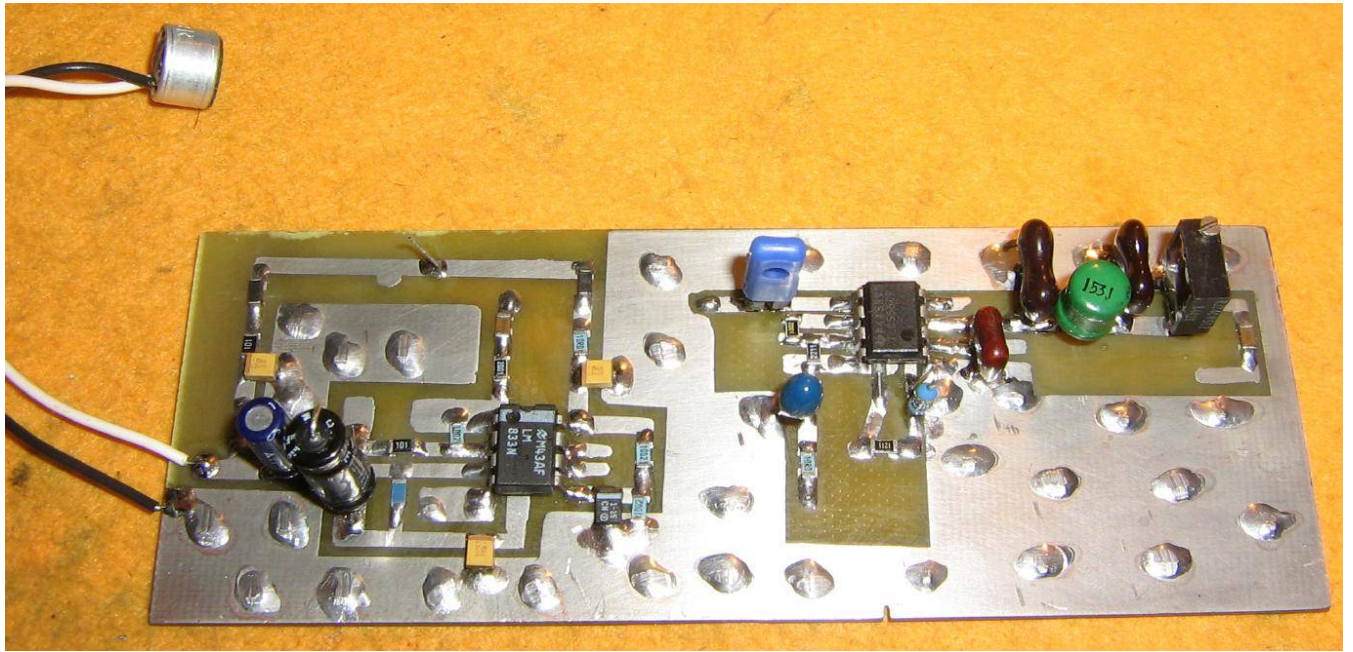
Overview of the 67 kHz frequency modulated subcarrier generator circuit.

A LM833−based microphone preamplifier is on the left.

It provides around 40 dB of gain and audio is from a standard electret microphone. The extra op−amp on the LM833 is used as an active bias to aid in circuit stability.

The 555 timer subcarrier generator is on the right. Its output frequency is set by a few resistors and a single high−quality 1000 pF capacitor. The audio output from the LM833 microphone preamplifier is fed to pin 5 on the 555 timer. This then frequency modulates the 67 kHz carrier generated by the 555 timer.

The entire circuit should be run from a well regulated +5 VDC power supply.

Use 1% metal−film resistors throughout the circuit for maximum low−noise performance and frequency stability.

This was a just a test circuit so it was physically quite large to allow for easier experimentation. Using SMT components will reduce the size of the circuit dramatically.
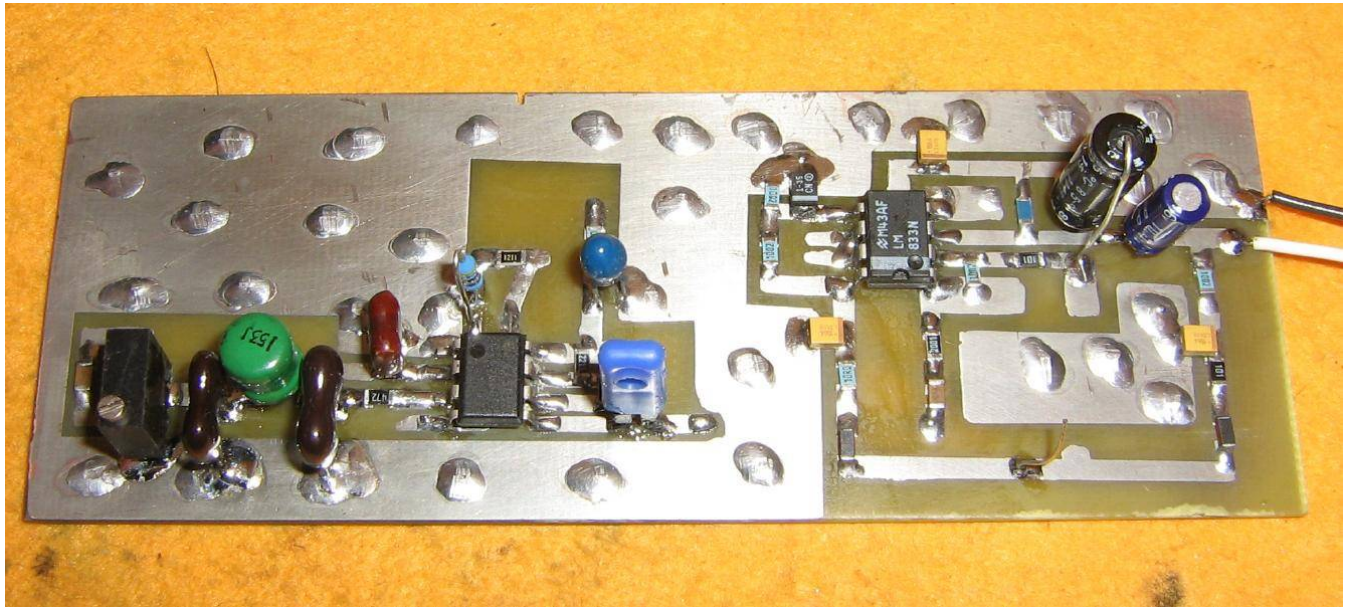
Alternate view of the subcarrier generator circuit.

On the output of the 555 timer (lower−left) is a 3−pole low−pass filter to remove any harmonics created by the subcarrier generator.
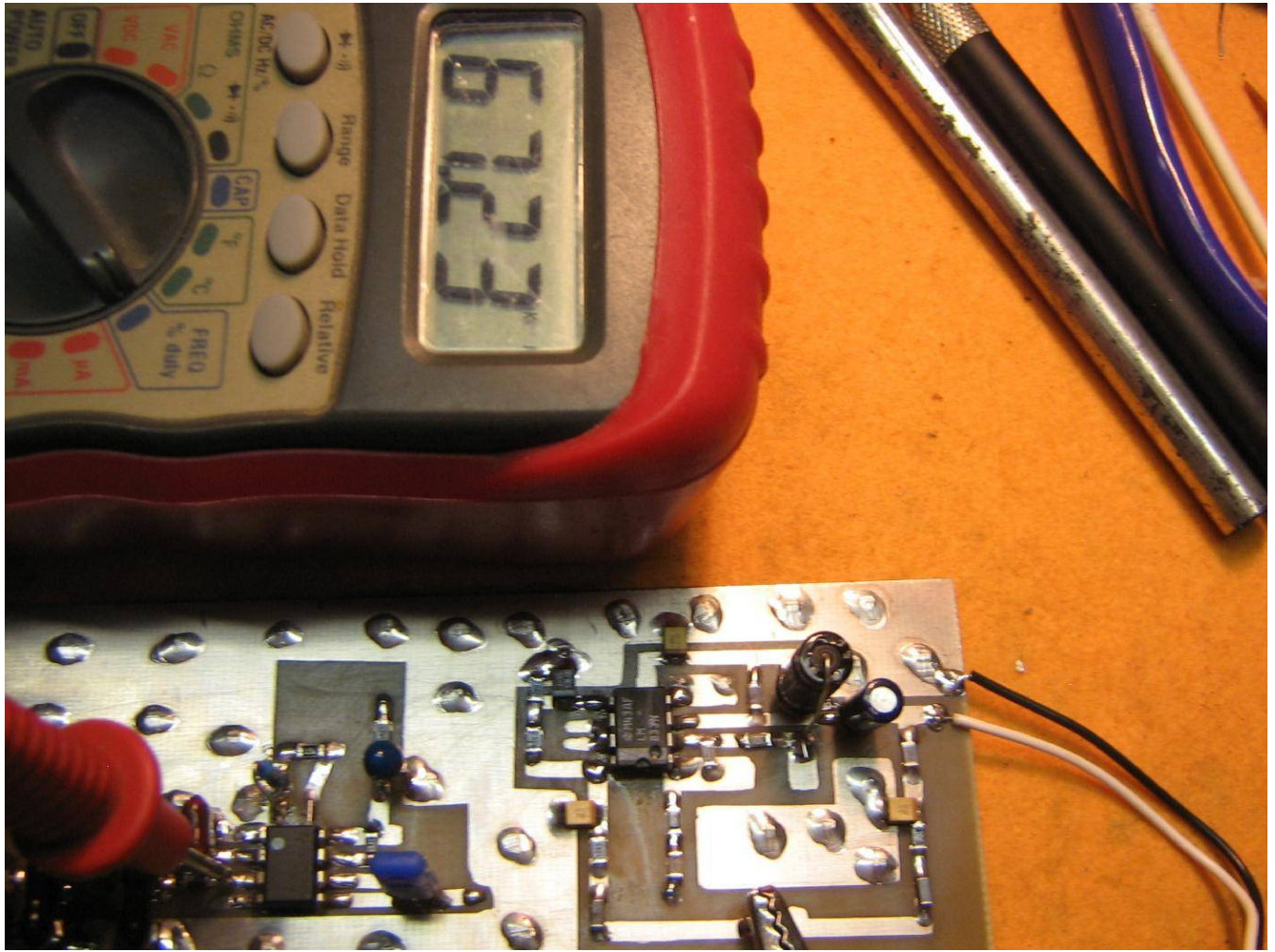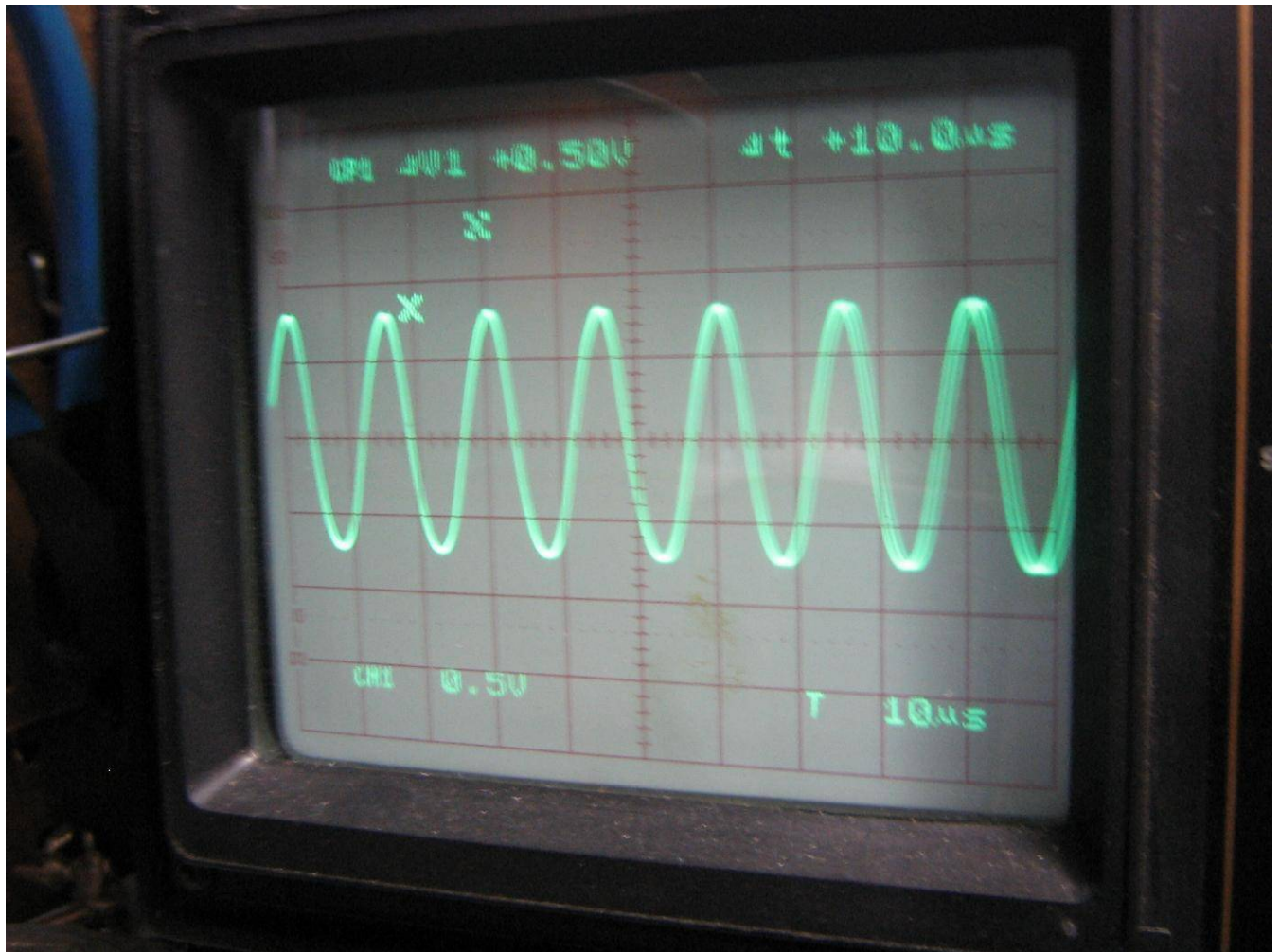
Tuning the frequency of the 555 timer subcarrier generator.

Monitor the output frequency on pin 3 of the 555 timer.

A multiturn 10 kohm potentiometer may be used to tweak the final output frequency.

The 555 timer generates a square wave, so a low–pass filter needs to be added on its output to remove any harmonics which may cause audio distortion in the transmitter.

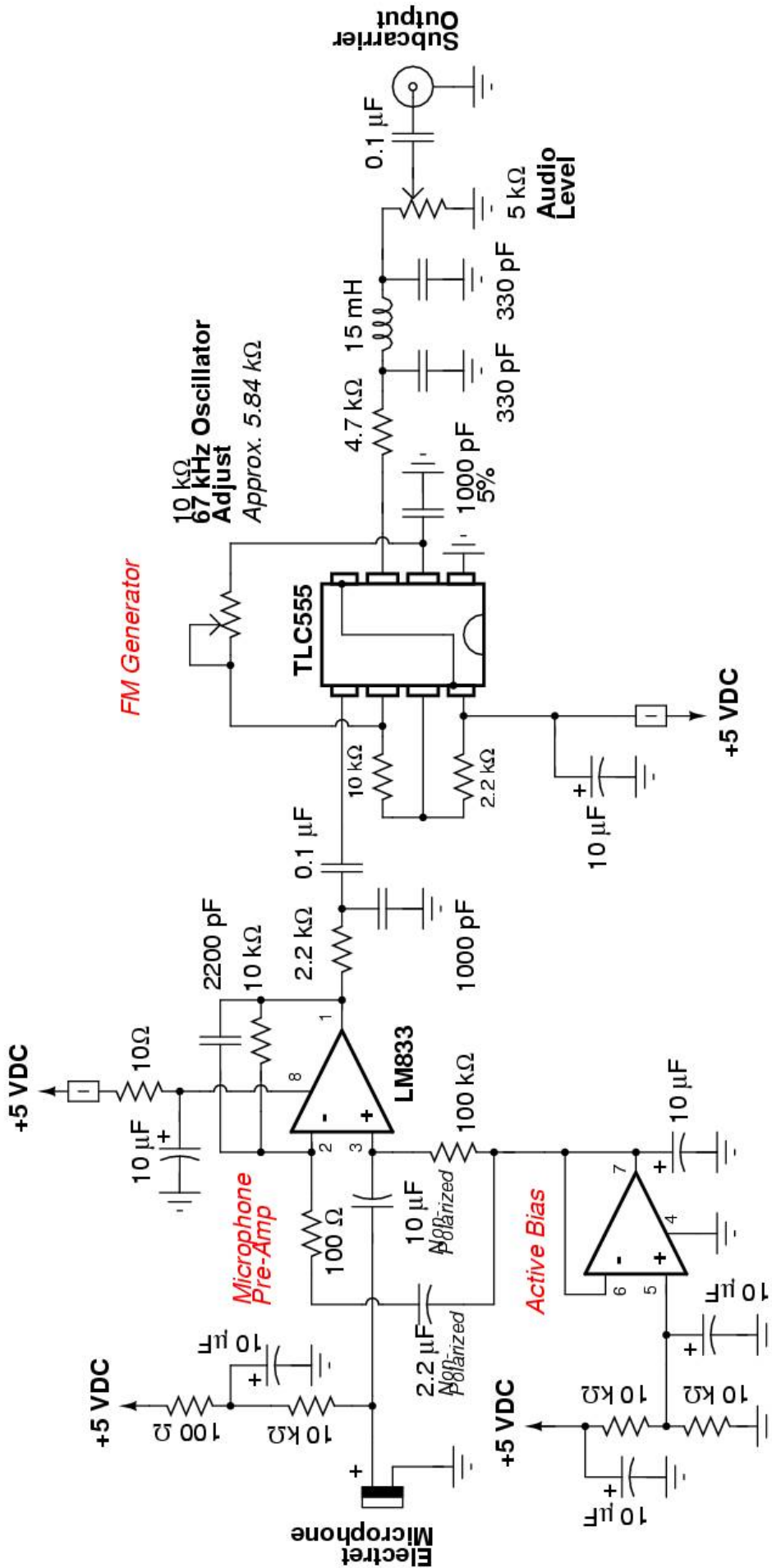Oscilloscope view of the final subcarrier output signal.

The low−pass filter turns the square wave output of the 555 timer into a sine wave.

The **Audio Level** adjustment potentiometer then further reduces the signal level before being applied to the modulation input on the transmitter.

Be sure not to over modulated the subcarrier generator, otherwise it will cause interference or splatter on nearby frequencies.

The subcarrier generator should be narrowband modulated (5 kHz deviation), while the actual RF transmitter may be wideband modulated (75 kHz deviation).

FM Subcarrier Generator

FM Generator

Subcarrier Output

0.1 μF

5 kΩ
Audio Level

330 pF

330 pF

4.7 kΩ 15 mH

10 kΩ
67 kHz Oscillator Adjust
Approx. 5.84 kΩ

TLC555

1000 pF 5%

10 kΩ

2.2 kΩ

+5 VDC

10 μF

0.1 μF

1000 pF

2200 pF

10 kΩ

2.2 kΩ

100 kΩ

1

8

LM833

2

3

+5 VDC

10 Ω

10 μF

Microphone Pre-Amp

100 Ω

10 μF
Non-Polarized

2.2 μF
Non-Polarized

Active Bias

10 μF

7

4

6

5

10 μF

+5 VDC

10 kΩ

100 Ω

10 μF

Electret Microphone

+5 VDC

10 kΩ

10 kΩ

10 μF

43

FREQUENCY MULTIPLICATION

There are two methods by which frequency multiplication can be achieved using the 565:

1. Locking to a harmonic of the input signal.
2. Inclusion of a digital frequency divider or counter in the loop between the VCO and phase comparator.

The first method is the simplest, and can be achieved by setting the free-running frequency of the VCO to a multiple of the input frequency. A limitation of this scheme is that the lock range decreases as successively higher and weaker harmonics are used for locking. If the input frequency is to be constant with little tracking required, the loop can generally be locked to any one of the first 5 harmonics. For higher orders of multiplication, or for cases where a large lock range is desired, the second scheme is more desirable. An example of this might be a case where the input signal varies over a wide frequency range and a large multiple of the input frequency is required.

A block diagram of the second scheme is shown in Figure 3. Here the loop is broken between the VCO and the phase comparator, and a frequency divider is inserted. The funda-
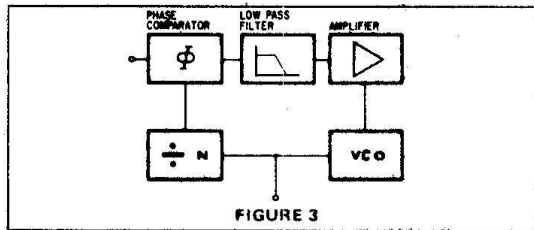

FIGURE 3

mental of the divided VCO frequency is locked to the input frequency in this case, so that the VCO is actually running at a multiple of the input frequency. The amount of multiplication is determined by the frequency divider. A typical connection scheme is shown in Figure 4. To set up the circuit, the frequency limits of the input signal must be determined. The free-running frequency of the VCO is then adjusted by means of R_1 and C_1 (as discussed under FM demodulation) so that the output frequency of the divider is midway between the input frequency limits. The filter capacitor, C_2, should be large enough to eliminate variations in the demodulated output voltage (at pin 7), in order to stabilize the VCO frequency. The output can now be taken as the VCO squarewave output, and its fundamental will be the desired multiple of the input frequency (f_1) as long as the loop is in lock.
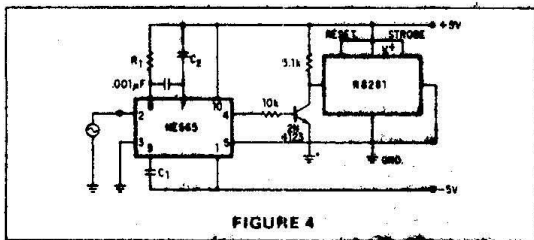

FIGURE 4

SCA (BACKGROUND MUSIC) DECODER

Some FM stations are authorized by the FCC to broadcast uninterrupted background music for commercial use. To do this a frequency modulated subcarrier of 67 kHz is used. The frequency is chosen so as not to interfere with the normal stereo or monaural program; in addition, the level of the subcarrier is only 10% of the amplitude of the combined signal.

The SCA signal can be filtered out and demodulated with the NE565 Phase Locked Loop without the use of any resonant circuits. A connection diagram is shown in Figure 5. This circuit also serves as an example of operation from a single power supply.

A resistive voltage divider is used to establish a bias voltage for the input (pins 2 and 3). The demodulated (multiplex) FM signal is fed to the input through a two-stage high-pass filter, both to effect capacitive coupling and to attenuate the strong signal of the regular channel. A total signal amplitude, between 80 mV and 300 mV, is required at the input. Its source should have an impedance of less than 10,000 ohms.

The Phase Locked Loop is tuned to 67 kHz with a 5000 ohm potentiometer; only approximate tuning is required, since the loop will seek the signal.

The demodulated output (pin 7) passes through a three-stage low-pass filter to provide de-emphasis and attenuate the high-frequency noise which often accompanies SCA transmission. Note that no capacitor is provided directly at pin 7; thus, the circuit is operating as a first-order loop. The demodulated output signal is in the order of 50 mV and the frequency response extends to 7 kHz.
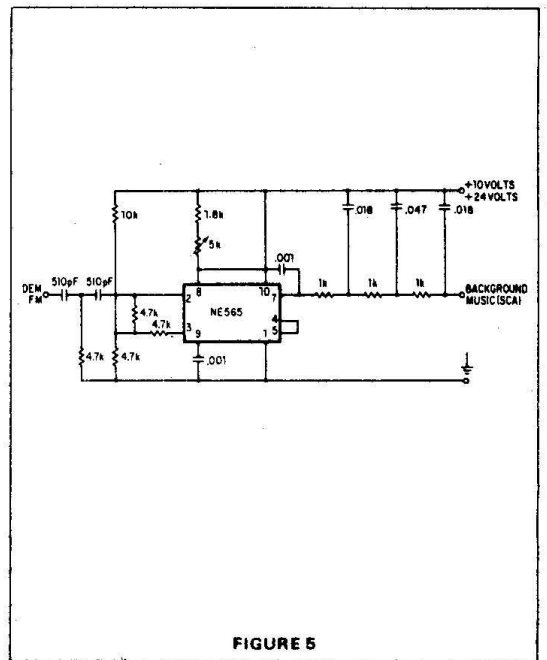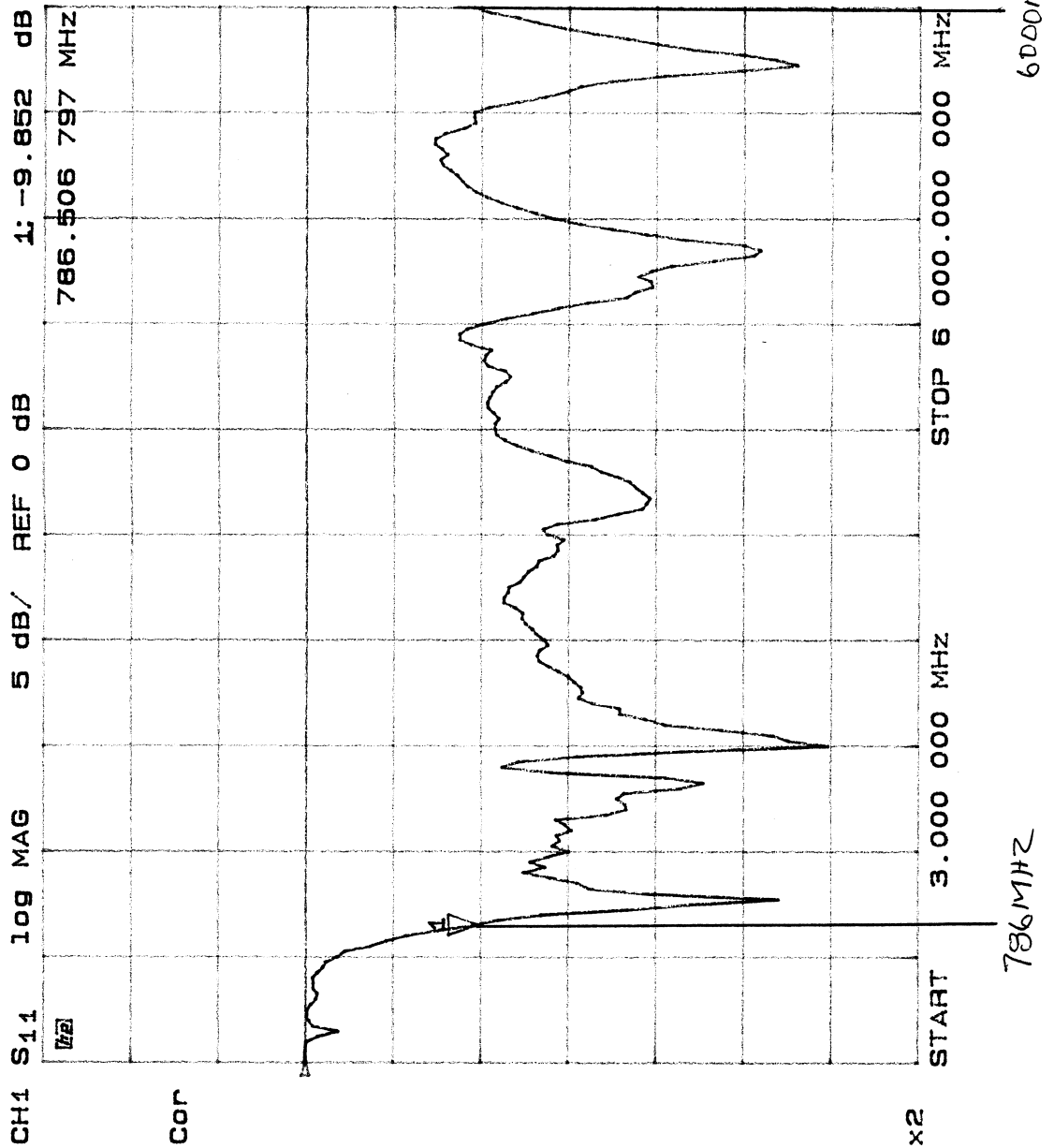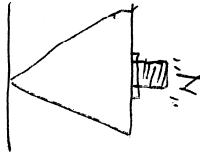

FIGURE 5

Discone Ant.
800MHz to 6000 MHz

CH1 S₁₁   log MAG   5 dB/   REF 0 dB   1: -9.852 dB
786.506 797 MHz

Cor

x2

START 3.000 000 MHz   STOP 6 000.000 000 MHz

786MHz   6000MHz

$4 a gallon gas. No protests...

SWAT teams raiding dairy farms for the milk they sell. No protests...

Federal Reserve is creating billions (if not trillions) out of thin air. No protests...

Never–ending wars for Israel's benefit. No protests...

The entire U.S. is slowing becoming a police state. No protests...

A failing public school system in more ways than one. No protests...

A manipulative media which covertly crafts your beliefs. No protests...

Massive fraud and corruption in the Obama administration. No protests...

A bunch of dumb niggers lose a football game because they were too stupid to score enough points to cover for any human errors. **PROTEST!**

Any Questions?

Editorial and Rants

Gotta' love stories like this...

Azmel Hussain, pictured, is defending the artwork which has been painted on his property. He described the wall as showing world leaders playing monopoly on a table held up by tax payers.

The mural, painted on a wall in Brick Lane, London, has been branded as "anti–Semitic" by Tower Hamlets council.

Banking Protest Mural Resembling Nazi anti–Semitic Propaganda to be Removed:
dailymail.co.uk/news/article–2213536/Banking–protest–mural–resembling–Nazi–anti–Semitic–propaganda
–removed–East–End.html

Jews/Cultural Marxists are always pushing degenerative art and "graffiti" onto the masses, but when someone makes something which hits a little too close to home – they run around screaming "censor!" LOL! Change!

The Department of Homeland Security wants to purchase some new ammo. O.K. No problems with that... But wait a minute! The DHS censored parts of the solicitation request! *Hmmm...* That's kinda suspicious...

Note it appears they used a marker to redact parts of the document before they scanned it. If you view the original PDF file and "zoom" in on the redacted parts – you can make out a few letters and words. Some PDF/graphics wiz should look into this some more...

(www.fbo.gov/utils/view?id=4c35c09c9315981b2f82beee2fc6629d)
(infowars.com/dhs–classifies–ammo–purchase–following–controversy)

JUSTIFICATION FOR OTHER THAN FULL AND OPEN COMPETITION
41 U.S.C. 3304 (a) (2)

Pursuant to the requirements of the U.S. Code 41 U.S.C. 3304 (a)(2) the Competition in Contracting Act (CICA) as implemented by Federal Acquisition Regulation (FAR), Subpart 6.3. and in accordance with the requirements of FAR, Subpart 6.302-2, the justification for the use of the statutory authority under FAR, Subpart 6.3 is justified by the following facts and rationale required under FAR Subpart 6.303-2 as follows:

1. Agency and Contracting Activity

The Department of Homeland Security (DHS), U.S. Immigration and Customs Enforcement (ICE), Office of Acquisition Management (OAQ) proposes to enter into a contract on a basis other than full and open competition.

2. Nature and/or Description of the Action Being Approved

ICE has a requirement to acquire .223 62 grain and 64 grain Duty ammunition for its own agents and officers, U.S. Customs and Border Protection (CBP) and the Federal Protective Service (FPS). Agents and officers from ICE, CBP and FPS will use the identified ammunition in the performance of their duties and for training as required by agency policy. ████████████████ ████████████████████████ The required ammunition is manufactured by Federal Cartridge Company, 900 Ehlen Drive, Anoka, MN 55303-1778.

3. Description of Supplies/Services

The contractor shall provide ammunition for ICE, CBP and FPS as specified below:

████████.223 62 grain duty ammunition████████
████████.223 64 grain duty ammunition████████

████████████████████████████ Initial delivery will be made 45 days after contract award with subsequent deliveries to be made at scheduled intervals throughout the 12 month period of performance.

4. Identification of Statutory Authority Permitting Other Than Full and Open Competition

The statutory authority permitting other than full and open competition is 41 U.S.C. 3304 (a)(2) implemented by the Federal Acquisition Regulation (FAR) Subpart 6.302-2 entitled "Unusual and Compelling Urgency." The manufacturer is the only one that provides the referenced part numbers for this one time purchase.

1

5. Demonstration That the Nature of the Acquisition Requires the Use Authority Cited

Based upon an assessment of the components stockpile for the ammunition to be acquired CBP and FPS have exhausted their stockpile supply. In addition, ICE has approximately 3 months of stockpile supplies of .223 (62 grain and 64 grain) duty ammunition. The failure of law enforcement agents being able to effectively perform their jobs could cause substantial safety issues for the government. Furthermore, the government would be severely impacted in its ability to deploy agents and officers to execute their missions which are an integral part of DHS operations. These products (.223 62 grain and 64 grain duty ammo) were previously obtained by ICE under IDIQ contracts ██████████████████████████████ The contracts recently expired and the results of the evaluation from the solicitation for new IDIQ contract resulted in no award for the standard duty ammunition. As a result, the National Firearms and Tactical Training Unit (NFTTU) are running the risk of depleting mission critical inventory.

██
██
██
███████████████

6. Description of Efforts Made to Ensure that Offers are Solicited from as Many Sources as is Practical

A Sources Sought was posted in FBO website (FedBizOpps) on May 7, 2012 in order to solicit for vendors that can provide the required ammo (.223 62 grain and 64 grain duty ammo). Even though small businesses expressed an interest in providing the required ammunition in response to the FBO posting, the specified part numbers can only be obtained from this manufacturer. In addition, the selected contractor has already met extensive testing requirements conducted by the ballistics laboratory (BALLAB).

7. Determination by the Contracting Officer that the Anticipated Cost to the Government will be Fair and Reasonable

The Contracting Officer determines that the anticipated pricing associated with .223 62 grain ██ ██████ and .223 64 grain ██████████ duty ammo will be fair and reasonable based on historical purchases. ██ These IDIQ contracts represent an excellent value to the Government, but has since expired and a new contract for .223 duty ammos is underway.

8. Description of Market Research

The NFTTU has conducted extensive market research in support of previously executed contracts ██████████████████████████████ These IDIQ contracts were awarded based on quality and price reasonableness. It is the intent of ICE to move forward with a sole source contract based on the urgent and compelling need for the ammunition.

2

Redacted Text:

```
P/N: XM223SP1  (Federal part number)
P/N: CS24450   (Speer part number)
These products have been utilized under prior government contract
```

9. Any Other Facts Supporting the Use of Other than Full and Open Competition

Conducting this procurement under full and open competition will delay the award and cause significant impact on the mission of ICE, CBP and FPS in terms of operational and training capabilities. The .223 ██ ████████████████████████████ duty ammo are made based on ICE specifications and BALLAB recommendations.

10. A Listing of the Sources, if any, that Expressed In Writing an Interest in the Acquisition

Vendors	Business Type
Ammo Broker	WOSB
PCP Ammunition	SB
Government Logistics Solution	SDVSOB
BCI Services	SDVOSB

11. A Statement of the Actions, if Any, the Agency May Take to Remove or Overcome Any Barriers to Competition Before Any Subsequent Acquisition for Supplies or Services Required

In the future, if additional .223 (62 grain and 64 grain) duty ammo is required before a replacement IDIQ contract can be awarded, market research will be performed to see if any other brands would be compatible and can be safely used to endure maximum competition.

12. Contracting Officer's Certification. I certify that the data supporting the recommended use of other than full and open competition is accurate and complete to the best of my knowledge.

████████████████████   ████████████

Contracting Officer   Date

13. Technical/Requirements Personnel Certification. I certify that this requirement meets the Government's minimum need and the supporting data, which forms a basis for this justification, is complete and accurate.

████████████████████   ████████████

Section Chief - Contracts   Date

3